DOCUMENT RESUME

ED 038 307 24 SE 008 254

AUTHOR	Kriewall, Thomas E.
TITLE	Applications of Information Theory and Acceptance Sampling Principles to the Management of Mathematics Instruction, Part 2.
INSTITUTION	Wisconsin Univ., Madison. Research and Development Center for Cognitive Learning.
SPONS AGENCY	Office of Education (DHEW), Washington, D.C. Bureau of Research.
REPORT NO	TR-103
BUREAU NO	BR-5-0216
PUB DATE	Oct 69
CONTRACT	OEC-5-10-154
NOTE	121p.
EDRS PRICE	EDRS Price MF-$0.50 HC-$6.15
DESCRIPTORS	*Computer Assisted Instruction, Doctoral Theses, *Individualized Instruction, *Information Theory, *Learning, Mathematics Education, *Research

ABSTRACT

This Technical Report is concerned with various problems of instructional management encountered in situations which stress self-selection and self-pacing principles. These problems deal primarily with the efficient utilization and allocation of human and material resource materials to formulate an operational, individualized, inquiry-learning environment. The specific purpose of this report is to develop a decision system based on a relevant theory of criterion-referenced tests and explore its potential for solving some problems which arise in the management of individualized mathematics curricula. The Report is a part of the Project on Computer-Managed Systems of Mathematics Instruction. Part II of this Report contains the Appendices which accompany research reported in Part I. (FL)

ED038307

BR 5-021
PA-24

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
OFFICE OF EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL OFFICE OF EDUCATION POSITION OR POLICY.

. 103

APPLICATIONS OF INFORMATION THEORY AND ACCEPTANCE SAMPLING PRINCIPLES TO THE MANAGEMENT OF MATHEMATICS INSTRUCTION

PART II

Report from the Project on
omputer-Managed Systems of Mathematics Instruction

SE 008 254

U.S. Office of Education Center No. C-03 Contract OE 5-10-154

ED038307

MAR 3 0 1970

Technical Report No. 103

APPLICATIONS OF INFORMATION THEORY AND ACCEPTANCE SAMPLING PRINCIPLES TO THE MANAGEMENT OF MATHEMATICS INSTRUCTION

PART II

Report from the Project on
Computer-Managed Systems of Mathematics Instruction

By Thomas E. Kriewall, Project Director

M. Vere DeVault, Professor of Curriculum and Instruction,
Chairman of the Examining Committee, and Principal Investigator

Wisconsin Research and Development
Center for Cognitive Learning
The University of Wisconsin
Madison, Wisconsin

October 1969

Published by the Wisconsin Research and Development Center for Cognitive Learning, supported in part as a research and development center by funds from the United States Office of Education, Department of Health, Education, and Welfare. The opinions expressed herein do not necessarily reflect the position or policy of the Office of Education and no official endorsement by the Office of Education should be inferred.

Center No. C-03 / Contract OE 5-10-154

NATIONAL EVALUATION COMMITTEE

Samuel Brownell
Professor of Urban Education
Graduate School
Yale University

Launor F. Carter
Senior Vice President on Technology and Development
System Development Corporation

Francis S. Chase
Professor
Department of Education
University of Chicago

Henry Chauncey
President
Educational Testing Service

Martin Deutsch
Director, Institute for Developmental Studies
New York Medical College

Jack Edling
Director, Teaching Research Division
Oregon State System of Higher Education

Elizabeth Koontz
President
National Education Association

Roderick McPhee
President
Punahou School, Honolulu

G. Wesley Sowards
Director, Elementary Education
Florida State University

Patrick Suppes
Professor
Department of Mathematics
Stanford University

***Benton J. Underwood**
Professor
Department of Psychology
Northwestern University

UNIVERSITY POLICY REVIEW BOARD

Leonard Berkowitz
Chairman
Department of Psychology

Archie A. Buchmiller
Deputy State Superintendent
Department of Public Instruction

***James W. Cleary**
Vice Chancellor for Academic Affairs

Leon D. Epstein
Dean
College of Letters and Science

John Guy Fowlkes
Director
Wisconsin Improvement Program

Robert E. Grinder
Chairman
Department of Educational Psychology

H. Clifton Hutchins
Chairman
Department of Curriculum and Instruction

Clauston Jenkins
Assistant Director
Coordinating Committee for Higher Education

Herbert J. Klausmeier
Director, R & D Center
Professor of Educational Psychology

Donald J. McCarty
Dean
School of Education

Ira Sharkansky
Associate Professor of Political Science

Henry C. Weinlick
Executive Secretary
Wisconsin Education Association

M. Crawford Young
Associate Dean
The Graduate School

EXECUTIVE COMMITTEE

Edgar F. Borgatta
Brittingham Professor of Sociology

Max R. Goodson
Professor of Educational Policy Studies

Russell J. Hosler
Professor of Curriculum and Instruction and of Business

***Herbert J. Klausmeier**
Director, R & D Center
Professor of Educational Psychology

Wayne Otto
Professor of Curriculum and Instruction (Reading)

Robert G. Petzold
Associate Dean of the School of Education
Professor of Curriculum and Instruction and of Music

Richard L. Venezky
Assistant Professor of English and of Computer Sciences

FACULTY OF PRINCIPAL INVESTIGATORS

Ronald R. Allen
Associate Professor of Speech and of Curriculum and Instruction

Vernon L. Allen
Associate Professor of Psychology
(On leave 1968-69)

Nathan S. Blount
Associate Professor of English and of Curriculum and Instruction

Robert C. Calfee
Associate Professor of Psychology

Robert E. Davidson
Assistant Professor of Educational Psychology

Gary A. Davis
Associate Professor of Educational Psychology

M. Vere DeVault
Professor of Curriculum and Instruction (Mathematics)

Frank H. Farley
Assistant Professor of Educational Psychology

John Guy Fowlkes (Advisor)
Professor of Educational Administration
Director of the Wisconsin Improvement Program

Lester S. Golub
Lecturer in Curriculum and Instruction and in English

Max R. Goodson
Professor of Educational Policy Studies

Warren O. Hagstrom
Professor of Sociology

John G. Harvey
Associate Professor of Mathematics and Curriculum and Instruction

Herbert J. Klausmeier
Director, R & D Center
Professor of Educational Psychology

Burton W. Kreitlow
Professor of Educational Policy Studies and of Agricultural and Extension Education

Richard G. Morrow
Assistant Professor of Educational Administration

Wayne Otto
Professor of Curriculum and Instruction (Reading)

Milton O. Pella
Professor of Curriculum and Instruction (Science)

Thomas A. Romberg
Assistant Professor of Mathematics and of Curriculum and Instruction

Richard L. Venezky
Assistant Professor of English and of Computer Sciences

MANAGEMENT COUNCIL

***Herbert J. Klausmeier**
Director, R & D Center
Acting Director, Program 1

Mary R. Quilling
Director
Technical Section

Thomas A. Romberg
Director
Programs 2 and 3

James E. Walter
Director
Dissemination Section

Dan G. Woolpert
Director
Operations and Business

* COMMITTEE CHAIRMAN

STATEMENT OF FOCUS

The Wisconsin Research and Development Center for Cognitive Learning focuses on contributing to a better understanding of cognitive learning by children and youth and to the improvement of related educational practices. The strategy for research and development is comprehensive. It includes basic research to generate new knowledge about the conditions and processes of learning and about the processes of instruction, and the subsequent development of research-based instructional materials, many of which are designed for use by teachers and others for use by students. These materials are tested and refined in school settings. Throughout these operations behavioral scientists, curriculum experts, academic scholars, and school people interact, insuring that the results of Center activities are based soundly on knowledge of subject matter and cognitive learning and that they are applied to the improvement of educational practice.

This Technical Report is from Phase 3 of the Project on Prototypic Instruction Systems in Elementary Mathematics in Program 2. General objectives of the Program are to establish rationale and strategy for developing instructional systems, to identify sequences of concepts and cognitive skills, to develop assessment procedures for those concepts and skills, to identify or develop instructional materials associated with the concepts and cognitive skills, and to generate new knowledge about instructional procedures. Contributing to the Program objectives, the Mathematics Project, Phase 1, is developing and testing a televised course in arithmetic for Grades 1-6 which provides not only a complete program of instruction for the pupils but also inservice training for teachers. Phase 2 has a long-term goal of providing an individually guided instructional program in elementary mathematics. Preliminary activities include identifying instructional objectives, student activities, teacher activities, materials, and assessment procedures for integration into a total mathematics curriculum. The third phase focuses on the development of a computer system for managing individually guided instruction in mathematics and on a later extension of the system's applicability.

TABLE OF CONTENTS

Part II

	Page
List of Tables	viii
List of Illustrations	ix
Acknowledgments	x
Abstract	xiii
I. PROBLEMS OF COMPUTER-FACILITATED MANAGEMENT IN MODERN MATHEMATICS CURRICULA	1-43
1.0 General Background and Nature of the Study	1-5
1.1 The Systems Approach: Basic Systems Concepts Applicable to Curricula	5-9
1.2 Development of A Systems Approach to Curriculum	9-12
1.3 Systems Concepts Applied to the Mathematics Curriculum: The State of the Art	12-14
1.4 Compatibility of a Systems Approach with Trends in Modern Mathematics Education	14-25
1.5 Problems of Individualizing Instruction in Mathematics	25-31
1.6 Problems of Educational Measurement	31-43
1.61 The Need for New Evaluation Metaphors	34-43
1.611 Item Difficulty	36-38
1.612 The Assumption of Normality	38-40
1.613 Test Reliability	41-43
II. MANAGEMENT SUPPORT SYSTEMS: THE MEASUREMENT INFORMATION COMPONENT	44-90
2.0 Introduction	44-45
2.1 Characteristics of Criterion-Tests	45-53
2.11 Absolute vs. Relative Measures	45-46
2.12 Content Validity	46-48
2.13 Parallel Item Selection	48-50
2.14 Minimal Test Length	50-51
2.15 Criterion Selection	51-53

CONTENTS (continued)

		Page
2.2	A Heuristic for Criterion-Referenced Tests . .	54-60
2.21	Assumptions of the Item-Sampling Model. .	55-57
2.22	Implications for Homogeneous Grouping . .	57-60
2.3	A Formal Theory of Criterion-Tests	60-80
2.31	Functions of A Theory	60-61
2.32	Order of A Model.	61-63
2.33	Basic Definitions for A First Order Theory.	63-64
2.331	Specified Content Objective. . . .	64-66
2.332	Proficiency and True Score	66-69
2.333	Criterion-Referenced Test	69-71
2.34	Sources of Measurement Error.	72-75
2.35	A CRT Performance Model	75-80
2.4	CRT Statistics	80-82
2.5	Implications of the CRT Model	82-87
2.51	For Item Selection.	82-84
2.52	For Test Reliability.	84-87
2.6	Comparison of the CRT Model with other Test Models	87-90
III.	MANAGEMENT SUPPORT SYSTEMS: THE DECISION COMPONENT	91-145
3.0	Introduction	91-95
3.01	The Role of Computer and Teacher in Semi-Automated Instructional Systems. . .	92-95
3.1	Process vs. Quality Control	95-99
3.2	Specification of Instructional Design Requirements	99-103
3.3	Specification of Acceptance Requirements . . .	103-108
3.4	Principles of Inspection Sampling of Learning Products	108-122
3.41	Test Outcomes as Points in a Probabilistic Sample Space	111-117
3.42	Scoring Formulas and Data Reduction . . .	117-119
3.43	Lattice Diagrams, CRT Performance Paths, and Criterion Boundaries	119-122

CONTENTS (continued)

		Page
3.5	Operating Characteristic of a Single Sampling Plan (SSP)	122-131
3.51	Ideal Sampling Plan Characteristics	123-125
3.52	Admissable Sampling Plan Characteristics	126-127
3.53	Impirical Considerations for Selecting An SSP	127-130
3.54	Procedures for Selecting An SSP	130-131
3.6	Curtailing the Single Sampling Plan	131-135
3.7	Decision-theoretic Considerations in Sampling Plan Selection	135-138
3.8	The Sequential Probability Ratio Test (SPRT)	138-141
3.9	Summary Comparison of Curtailed and Single Sampling.	141-145
IV.	IMPLICATIONS FOR CONTINUED PROGRAMMATIC RESEARCH	146-177
4.0	Overview	146-147
4.1	Implications for Further Development of Criterion-Reference Test Theory	147-165
4.11	Affective Validation	149-154
4.12	Functional Validation	154-163
4.13	Formal Validation	163-165
4.2	Implications for Curriculum and Instruction	166-175
4.21	Identifying Curriculum Hierarchies	166-167
4.22	Predictive Learning Theory	167-173
4.23	Implications for Assessing Instructional Effectiveness	173-175
4.3	Implications for the Systems Approach to Education	175-176
4.4	Implications for Teacher Education	176-177
Appendix A	Sample Criterion Tests and Test Results	178-186
Appendix B	Sample Computer-Generated Criterion Tests	187-210
Appendix C	List of Symbols and Notation Used	211-213
Appendix D	Computer Programs for Generating OC and ASN Curves	214-219
Appendix E	Notes on the Sequential Probability Ratio Test	220-227
Appendix F	Prototypic Program and Data Files.	228-262
Appendix G	Figures 1-16	263-277
References		278-285

LIST OF TABLES

TABLE		PAGE
3-1	Specification of Design Requirements	100
3-2	Illustrative Design Requirements	111
3-3	Illustrative Acceptance Requirements	112
3-4	Probabilities Associated with Response Patterns and Test Scores	115
3-5	Comparison Data on Acceptance Requirements . . .	129

LIST OF ILLUSTRATIONS

FIGURE		PAGE
1	A General Model for the ESM Curriculum	264
2	A Pattern for Instruction	265
3	Lattice Diagram (n=3)	266
4	Lattice Diagram (n=20)	267
5	OC Curves for Fixed N and Selected Criteria (n=5)	268
6	OC Curves for Fixed N and Selected Criteria (n=20)	269
7	OC Curves for Fixed N and Selected Criteria (n=25)	270
8	Step Function OC.	271
9	OC with Region of Indifference.	271
10	Decision Outcome Probabilities.	271
11	ASN Curves for Fixed N and Selected Criteria (n=5)	272
12	ASN Curves for Fixed N and Selected Criteria (n=20)	273
13	ASN Curves for Fixed N and Selected Criteria (n=25)	274
14	Comparative Sample Sizes for Various Sampling Plans	275
15	Random Sample of 2-Digit Multiplication Problems Generated by MATH/CMS	276
16	Hypothetical Proficiency Development Curve.	277

ACKNOWLEDGMENTS

The task of properly acknowledging all who contributed to this work presents to the author perhaps the most difficult challenge of the entire effort. This is partly so because, in the case of Vere DeVault--my major professor, inspirer, and guide--no expression of thanks would seem fully adequate. In the case of other committee members, particularly Frank Baker and Anne Cleary, their persistent and perceptive probing led not only to a better final product but also, along the way, to some energetic exchanges concerning our differences in regard to the sum and substance of educational measurement. An expression of gratitude to both is in order and hereby tendered, but there lingers the feeling that possibly touché would be more appropriate than merci.

The contributions of others have been made in perhaps less dramatic and certainly less traumatic form. J. Fred Weaver, Al Yee, and Dan Anderson have, over the many months of this investigation, contributed in a variety of ways for which I wish to express my appreciation. Harry Silberman and John Coulson of Systems Development Corporation provided the author with a unique opportunity to work on the criterion-test theory. To James Bavry of the Wisconsin Research and Development Center goes my thanks for working out the computer programming needed to graph the operating characteristics and average sample number associated with the sequential

probability ratio test. Others at the R and D Center who were particularly helpful, especially in regard to measurement questions, were Tom Romberg and Tom Fishbach.

My appreciation is also expressed for the excellent work done by the artists, publication and technical personnel, and secretaries of the Center all of whom met the challenge of the Greek alphabet and my penmanship with unfailing good grace and impressive competence. Special mention is merited by the fine work performed by our project secretary, Ethel Koshalek; the thesis typists Jan Sluga, Dorothy Egener, and Darlene Barkema; and to Doris Ardelt and Julie Erjavec who together managed the final production of the paper.

Use of the University of Wisconsin Computing Center was made possible through support, in part, from the National Science Foundation, other United States Government agencies and the Wisconsin Alumni Research Foundation (WARF) through the University of Wisconsin Research Committee.

This work was made possible through the facilities of the Wisconsin Research and Development Center for Cognitive Learning, Herbert J. Klausmeier, Director.

ABSTRACT

Individualization of instruction and the development of inquiry learning techniques are two major areas of interest in mathematics education today. Those teachers who would implement individualized programs of instruction and would encourage pupils to take increasing responsibility for inquiring about mathematics objectives of their own choosing face many problems of classroom management, data acquisition, record-keeping, diagnosing, prescribing, and decision-making for which traditional classroom management practices are poorly suited. This dissertation treats several problems of instructional management encountered in situations which emphasize self-selection and self-pacing principles. Primarily, these problems deal with the efficient utilization and allocation of available human and material resources to create an operational, individualized, inquiry-learning environment.

The systems approach to education is first explored to determine the extent to which systems disciplines, such as utility theory and operations analysis, can be applied to facilitate classroom management and decision-making procedures. This exploration leads to an examination of test-design, since tests comprise the basic means of obtaining decision-data. Traditional, norm-referenced test theory is shown to

suffer a number of disadvantages from the instructional management viewpoint. Therefore, an approach toward test design is developed based on certain principles of item-sampling and criterion-testing.

The criterion-test is seen as a sequence of independent Bernoulli trials on items randomly selected from a well-defined population of items. The item pool is operationally defined by a Specified Content Objective or, equivalently, by a set of item-generation rules. This approach leads to a binomial test model that is utilized in the design of acceptance sampling procedures. These procedures minimize testing time required to obtain suitably reliable pupil proficiency profiles.

A prototypic computer-managed system of mathematics instruction demonstrates the use of criterion-referenced test theory and acceptance sampling principles. The prototype is restricted for demonstration purposes to the universe of integers, numeration systems two through ten, simple equivalence statements, and the four fundamental arithmetic operations. Data gathering, decision-making, and information handling techniques are also illustrated by the program and the protocols involved in using its associated data files.

Suggestions are offered for reducing the testing time needed to detect mastery attainment levels consistent with Neyman-Pearson theory. The relationship between mastery criteria and such sampling plans as single sampling, simple curtailed testing, and the use of the sequential probability ratio test is discussed. Applications are indicated in the area of computer-generation of test items and automated administration of criterion-referenced tests of mastery in selected arithmetic skills.

The use of criterion-referenced test results, evaluated by sequential analytic techniques indicates promise for reducing testing time and costs for specified behavioral objectives. This, in turn, promises the possibility of designing improved and extended capabilities for computer-assisted instructional management systems.

APPENDIX A

APPENDIX A

Three sample criterion-referenced pretests are shown on pages 179 and 180. Posttests containing items randomly sampled from corresponding pools are shown on 181 and 182. The "fail-safe" box simply permitted pupils who felt they had no proficiency whatever on a given st of problems to bypass the set without embarrassment. Pupils made use of this option mainly on pretest C.

Test results for a class of 19 fifth-graders are summarized on pages 183 to 187. The "high" group on each test consisted of pupils who made fewer than two errors. Test reliabilities are relatively high for the total group but become erratic when computed for the relatively homogeneous proficiency subgroups. This is simply an expected consequence of the fact that variation of scores within a subgroup is mostly error variation.

The observed proficiency gains and transition matrices are illustrative of the kind of management information that this type of criterion-referenced test may provide.

The OC curve for the sampling plan employed on these tests ($n = 5$; $c = 2$) is shown on Figure 5 (Appendix G). If the tests are expected to discriminate a maximum error rate of $\zeta_1' = .15$ for the high group and a minimum error rate of $\zeta_2' = .65$ for the low group, the OC curve shows that the errors of classification will be approximately $\alpha = .16$ and $\beta = .04$ for this plan. Shifting to an error criterion of $c = 1$ would greatly increase α but only slightly decrease β.

Name _____

Instructions: There are three parts on this test. If you decide you don't know how to do the problems in any part, put an X in the "fail-safe" box and go on to the next part.

Part A

Fail-safe: ☐

For each fraction, find the equivalent fraction in lowest terms.

EXAMPLE:

Problem	Response
$\frac{16}{40}$ =	$\frac{2}{5}$

Problem	Response	Problem	Response	Problem	Response
1. $\frac{10}{36}$ =		3. $\frac{27}{60}$ =		5. $\frac{35}{42}$ =	
2. $\frac{18}{48}$ =		4. $\frac{12}{14}$ =			

Part B

Fail-safe: ☐

For each fraction, find the equivalent mixed fraction in lowest terms.

EXAMPLE:

Problem	Response
$\frac{30}{12}$ =	$2\frac{1}{2}$

Problem	Response	Problem	Response	Problem	Response
6. $\frac{42}{12}$ =		8. $\frac{70}{36}$ =		10. $\frac{63}{25}$ =	
7. $\frac{63}{54}$ =		9. $\frac{42}{9}$ =			

Part C

Fail-safe: ☐

Find the sums.

EXAMPLE:

Problem	Solution	Response
$\frac{1}{6} + \frac{4}{5}$	$= \frac{5}{30} + \frac{24}{30}$	$= \frac{29}{30}$

	Problem	Solution	Response
11.	$\frac{3}{4} + \frac{1}{8} =$		
12.	$\frac{3}{5} + \frac{2}{7} =$		
13.	$\frac{0}{4} + \frac{6}{7} =$		

	Problem	Solution	Response
14.	$\frac{5}{7} + \frac{1}{2} =$		
15.	$\frac{1}{9} + \frac{4}{7} =$		

Name _____

Instructions: There are three parts on this test. If you decide you don't know how to do the problems in any part, put an X in the "fail-safe" box and go on to the next part.

Part A Fail-safe: ☐

For each fraction, find the equivalent fraction in lowest terms.

Problem	Response	Problem	Response	Problem	Response
1. $\frac{35}{50}$ =		3. $\frac{35}{49}$ =		5. $\frac{10}{28}$ =	
2. $\frac{15}{35}$ =		4. $\frac{21}{30}$ =			

Part B Fail-safe: ☐

For each fraction, find the equivalent mixed fraction in lowest terms.

Problem	Response	Problem	Response	Problem	Response
6. $\frac{50}{21}$ =		8. $\frac{70}{21}$ =		10. $\frac{98}{25}$ =	
7. $\frac{45}{35}$ =		9. $\frac{70}{54}$ =			

Part C

Fail-safe: ☐

Find the sums.

Problem	Solution	Response
11. $\frac{6}{7} + \frac{2}{9} =$		
12. $\frac{0}{9} + \frac{1}{2} =$		
13. $\frac{1}{5} + \frac{1}{4} =$		
14. $\frac{4}{9} + \frac{6}{7} =$		
15. $\frac{2}{7} + \frac{1}{4} =$		

DATA SUMMARY

Test Identification	Mean Proportion Correct			Group Size			KR-20		
	Group						Group		
	Total	High	Low	N	N High	N Low	Total	High	Low
Pretest A	.43	.88	.27	19	5	14	.74	-1.0	.28
B	.42	.85	.31	19	4	15	.70	-.02	.41
C	.00	.00	.00	19	0	19	-	-	-
Posttest A	.63	.98	.38	19	8	11	.82	.04	.48
B	.57	.85	.36	19	8	11	.74	-1.0	.48
C	.39	.92	.20	19	5	14	.88	0.0	.68

n = 5

c = 2

ESTIMATED PROFICIENCY GAIN

	Objective A	Objective B	Objective C
Total Group	.20	.15	.39
High Group	.10	.00	.92
Low Group	.11	.05	.20

TRANSITION MATRICES

Test A
From Level

To Level	High	Low	Final Totals
High	4	4	8
Low	1	10	11
Initial Totals	5	14	19

Test B
From Level

To Level	High	Low	Final Totals
High	3	5	8
Low	1	10	11
Initial Totals	4	15	19

Test C
From Level

To Level	High	Low	Final Totals
High	0	5	5
Low	0	14	14
Initial Totals	0	19	19

APPENDIX B

APPENDIX B

Tests included in this Appendix were produced by the program MATH/CMS using parameters read from the file STAN/CMS (cf. Appendix F). These tests illustrate three points discussed in the main body of the dissertation.

First, the tests are administered in batch mode, or what is called "Mode 0" operation in Chapter Four. A master copy is produced on the TTY by the computer and a Ditto Spirit Master is obtained by a heat transfer process. Copies of each test are duplicated, distributed to certain pupils, and collected by a teacher's aide when each student finishes the test. A coded key, matched to the test by the Test Number (which includes the starting number for the random number generator as the last eight digits), is used by the aide to hand-score the tests. Information is returned to the computer using the Systems program EDIT. This test data then can be used to form homogeneous proficiency groups. Rosters of group members are returned either to the TTY or printed on the on-line Printer at the Computing Center as desired. Printouts are mailed the same day test data is received, in the latter case.

The tests also illustrate the use of the SSP. Sample size is fixed, under parametric control, as is the criterion for separating pupils into relatively homogeneous proficiency groups. This technique represents the most economical use of computer and terminal although further study is needed to determine its comparative efficiency with more highly interactive sampling plans, such as the SPRT.

Finally, random generation of items from a well-defined population, or SCO, is exemplified in each test. If each item is regarded as a stimulus, then all stimuli are "memorized" in the 8-digit starting number of the random number generator. (The digits preceding the starting number comprise the sequential test number, or record number, by which the parameters are accessed in file STAN/CMS.) Thus any test can be exactly reproduced at a later time, if desired, by using Mode 3 and the starting number. It may also be noted that random sampling of items assist teachers in the problem of maintaining effective test security. More importantly, it provides an economical method of producing reliable, parallel tests where many may be needed in a management system such as that described in this paper.

TEST # 175863844 NAME _____

DATE _____

ALL PROBLEMS ARE BASE 10:

1). 0 + 5 = 2). 5 + 6 =

3). 8 + 1 = 4). 2 + 5 =

5). 2 + 7 = 6). 7 + 7 =

7). 3 + 7 = 8). 1 + 0 =

9). 8 + 2 = 10). 7 + 4 =

BASIC ADDITION FACTS (0-9): TEST #1.
ANSWER KEY: TEST # 175863844

PROBLEM ANSWER REMAINDER

1	5
2	11
3	9
4	7
5	9
6	14
7	10
8	1
9	10
10	11

IF FINISHED, ENTER 0 ELSE ENTER 1
:

TEST # 215786720 NAME _____

DATE _____

ALL PROBLEMS ARE BASE 10:

1). 14 - 9 =

2). 9 - 2 =

3). 6 - 4 =

4). 13 - 9 =

5). 5 - 2 =

6). 10 - 4 =

7). 10 - 4 =

8). 11 - 9 =

9). 13 - 5 =

10). 18 - 9 =

BASIC SUBTRACTION FACTS (0-9): TEST #2.
ANSWER KEY: TEST # 215786720

PROBLEM	ANSWER	REMAINDER
1	5	
2	7	
3	2	
4	4	
5	3	
6	6	
7	6	
8	2	
9	8	
10	9	

IF FINISHED, ENTER 0 ELSE ENTER 1
:

TEST # 392074756 NAME _____

DATE _____

ALL PROBLEMS ARE BASE 10:

1). 4 X 9 =

2). 7 X 7 =

3). 0 X 9 =

4). 7 X 2 =

5). 6 X 5 =

6). 0 X 5 =

7). 2 X 5 =

8). 3 X 8 =

9). 0 X 1 =

10). 8 X 6 =

BASIC MULTIPLICATION FACTS (0-9): TEST #3.
ANSWER KEY: TEST # 392074756

PROBLEM	ANSWER	REMAINDER
1	36	
2	49	
3	0	
4	14	
5	30	
6	0	
7	10	
8	24	
9	0	
10	48	

IF FINISHED, ENTER 0 ELSE ENTER 1
:

TEST # 413629632 NAME _____

DATE _____

ALL PROBLEMS ARE BASE 10:

1).	21 / 3 =	2).	0 / 6 =
3).	8 / 8 =	4).	30 / 5 =
5).	9 / 1 =	6).	63 / 9 =
7).	54 / 6 =	8).	15 / 3 =
9).	6 / 3 =	10).	0 / 5 =

BASIC DIVISION FACTS (0-9): TEST #4.
ANSWER KEY: TEST # 413629632

PROBLEM	ANSWER	REMAINDER
1	7	
2	0	
3	1	
4	6	
5	9	
6	7	
7	9	
8	5	
9	2	
10	0	

IF FINISHED, ENTER 0 ELSE ENTER 1
:

TEST # 5 9072230 NAME _____

DATE _____

ALL PROBLEMS ARE BASE 10:

1).	2).	3).	4).
301	240	235	213
+ 174	+ 735	+ 632	+ 406
------	------	------	------

5).
143
+ 832

3-DIGIT ADDITION WITHOUT REGROUPING: TEST #5.
ANSWER KEY: TEST # 5 9072230

PROBLEM	ANSWER	REMAINDER
1	475	
2	975	
3	867	
4	619	
5	975	

IF FINISHED, ENTER 0 ELSE ENTER 1
:

TEST # 621844638 NAME _____

DATE _____

ALL PROBLEMS ARE BASE 10:

1).	2).	3).	4).
97	47	66	44
+ 23	+ 98	+ 97	+ 98
------	------	------	------

5).
76
+ 96

2-DIGIT ADDITION WITH REGROUPING: TEST #6.
ANSWER KEY: TEST # 621844638

PROBLEM	ANSWER	REMAINDER
1	120	
2	145	
3	163	
4	142	
5	172	

IF FINISHED, ENTER 0 ELSE ENTER 1
:

TEST # 712333058 NAME _____

DATE _____

ALL PROBLEMS ARE BASE 10:

1). 9141 + 2145	2). 3296 + 5188	3). 2963 + 1135	4). 2062 + 2126

5). 1453 + 3920

4-DIGIT ADDITION: TEST #7.
ANSWER KEY: TEST # 712333058

PROBLEM	ANSWER	REMAINDER
1	11286	
2	8484	
3	4098	
4	4188	
5	5373	

IF FINISHED, ENTER 0 ELSE ENTER 1
:

TEST # 854760530 NAME _____

DATE _____

ALL PROBLEMS ARE BASE 10:

1).	2).	3).	4).
53	14	26	69
35	62	37	52
45	16	98	30
+ 61	+ 60	+ 58	+ 86
------	------	------	------

5).
71
73
64
+ 45

2-DIGIT COLUMN ADDITION, 4 ADDENDS: TEST #8.
ANSWER KEY: TEST # 854760530

PROBLEM	ANSWER	REMAINDER
1	194	
2	152	
3	219	
4	237	
5	253	

IF FINISHED, ENTER 0 ELSE ENTER 1
:

TEST # 914626828 NAME _____

DATE _____

ALL PROBLEMS ARE BASE 10:

1).	2).	3).	4).
564	597	685	774
- 100	- 274	- 341	- 201
------	------	------	------

5). 573
- 253

3-DIGIT SUBTRACTION WITHOUT REGROUPING: TEST #9.
ANSWER KEY: TEST # 914626828

PROBLEM ANSWER REMAINDER

1	464
2	323
3	344
4	573
5	320

IF FINISHED, ENTER 0 ELSE ENTER 1
:

TEST # 1040253820 NAME _____

DATE _____

ALL PROBLEMS ARE BASE 10:

1).	2).	3).	4).
91	85	52	66
- 62	- 49	- 33	- 29
------	------	------	------

5).
50
- 27

2-DIGIT SUBTRACTION WITH REGROUPING: TEST #10.
ANSWER KEY: TEST # 1040253820

PROBLEM	ANSWER	REMAINDER
1	29	
2	36	
3	19	
4	37	
5	23	

IF FINISHED, ENTER 0 ELSE ENTER 1
:

TEST # 1111404569 NAME _____

DATE _____

ALL PROBLEMS ARE BASE 10:

1). 9381 - 1204 ------

2). 8491 - 4439 ------

3). 9567 - 5788 ------

4). 8335 - 5984 ------

5). 3824 - 1425 ------

4-DIGIT SUBTRACTION: TEST #11.
ANSWER KEY: TEST # 1111404569

PROBLEM ANSWER REMAINDER

1	8177
2	4052
3	3779
4	2351
5	2399

IF FINISHED, ENTER 0 ELSE ENTER 1
:

TEST # 1250696600 NAME _____

DATE _____

ALL PROBLEMS ARE BASE 10:

1). 111 X 3 = _____

2). 222 X 4 = _____

3). 302 X 3 = _____

4). 332 X 2 = _____

5). 701 X 1 = _____

3 BY 1 DIGIT MULTIPLICATION WITHOUT REGROUPING: TEST #12.
ANSWER KEY: TEST # 1250696600

PROBLEM	ANSWER	REMAINDER
1	333	
2	888	
3	906	
4	664	
5	701	

IF FINISHED, ENTER 0 ELSE ENTER 1
:

TEST # 1322366894 NAME _____

DATE _____

ALL PROBLEMS ARE BASE 10:

1). 56 X 75 ------

2). 79 X 67 ------

3). 53 X 75 ------

4). 57 X 88 ------

5). 99 X 29 ------

2 BY 2 DIGIT MULTIPLICATION WITH REGROUPING: TEST #13.
ANSWER KEY: TEST # 1322366894

PROBLEM	ANSWER REMAINDER
1	4200
2	5293
3	3975
4	5016
5	2871

IF FINISHED, ENTER 0 ELSE ENTER 1
:

TEST # 1414269738 NAME _____

DATE _____

ALL PROBLEMS ARE BASE 10:

1). 9126 X 348 ------

2). 3253 X 691 ------

3). 3751 X 730 ------

4). 8239 X 578 ------

5). 7066 X 669 ------

4 BY 3 DIGIT MULTIPLICATION: TEST #14.
ANSWER KEY: TEST # 1414269738

PROBLEM ANSWER REMAINDER

1 3175848
2 2247823
3 2738230
4 4762142
5 4727154

IF FINISHED, ENTER 0 ELSE ENTER 1
:

TEST # 1540872060 NAME _____

DATE _____

ALL PROBLEMS ARE BASE 10:

1). 2). 3). 4).

```
       -------        -------        -------        -------
    2 ) 668        5 ) 1055       4 ) 1200       2 ) 1622
```

5).

```
       -------
    3 ) 666
```

3 BY 1 DIGIT DIVISION WITHOUT REGROUPING: TEST #15.
ANSWER KEY: TEST # 1540872060

PROBLEM	ANSWER	REMAINDER
1	334	
2	211	
3	300	
4	811	
5	222	

IF FINISHED, ENTER 0 ELSE ENTER 1
:

TEST # 1640982140 NAME _____

DATE _____

ALL PROBLEMS ARE BASE 10:

1). 2). 3). 4).

```
      -------        -------        -------        -------
   3 ) 2444       2 ) 8019       3 ) 4078       6 ) 1282
```

5).

```
      -------
   3 ) 9392
```

4 BY 1 DIGIT DIVISION WITH REGROUPING: TEST #16.
ANSWER KEY: TEST # 1640982140

PROBLEM	ANSWER	REMAINDER
1	814	2
2	4009	1
3	1359	1
4	213	4
5	3130	2

IF FINISHED, ENTER 0 ELSE ENTER 1
:

TEST # 17 671088

NAME _____

DATE _____

ALL PROBLEMS ARE BASE 10:

1).
```
      -------
   2 ) 814
```

2).
```
      -------
   2 ) 606
```

3).
```
      -------
   3 ) 603
```

4).
```
      -------
   3 ) 321
```

5).
```
      -------
   3 ) 303
```

3 BY 1 DIGIT DIVISION,ZERO IN QUOTIENT: TEST #17.
ANSWER KEY: TEST # 17 671088

PROBLEM	ANSWER	REMAINDER
1	407	
2	303	
3	201	
4	107	
5	101	

IF FINISHED,ENTER 0 ELSE ENTER 1
:

TEST # 1868341119 NAME _____

DATE _____

ALL PROBLEMS ARE BASE 10:

1). 49) 588

2). 39) 429

3). 73) 876

4). 33) 957

5). 69) 690

3 BY 2 DIGIT DIVISION,NO REMAINDERS: TEST #18.
ANSWER KEY: TEST # 1868341119

PROBLEM	ANSWER	REMAINDER
1	12	
2	11	
3	12	
4	29	
5	10	

IF FINISHED,ENTER 0 ELSE ENTER 1
:

TEST # 1955773630 NAME _____

DATE _____

ALL PROBLEMS ARE BASE 10:

1). 140) 4846

2). 132) 9278

3). 792) 5426

4). 440) 3662

5). 911) 3287

DIVISION BY 3-DIGIT DIVISORS,REMAINDERS NOT ZERO: TEST #19.
ANSWER KEY: TEST # 1955773630

PROBLEM	ANSWER	REMAINDER
1	34	86
2	70	38
3	6	674
4	8	142
5	3	554

IF FINISHED,ENTER 0 ELSE ENTER 1
:←

TO START A NEW MODE ENTER 1 ELSE ENTER 0
:

TEST # 2418464136 NAME _____

DATE _____

ALL PROBLEMS ARE BASE 5:

1). 1 + 2 = ------

2). 0 + 4 = ------

3). 0 + 3 = ------

4). 3 + 1 = ------

5). 2 + 4 = ------

BASE FIVE ADDITION FACTS (0-4): TEST #24.
ANSWER KEY: TEST # 2418464136

PROBLEM	ANSWER	REMAINDER
1	3	
2	4	
3	3	
4	4	
5	11	

IF FINISHED, ENTER 0 ELSE ENTER 1
:

TEST # 2536059856 NAME _____

DATE _____

ALL PROBLEMS ARE BASE 5:

```
1).    11      2).     3      3).     3      4).     4
     - 3             - 0            - 2            - 3
     ------          ------         ------         ------

5).    10
     - 2
     ------
```

BASE FIVE SUBTRACTION FACTS (0-13): TEST #25.
ANSWER KEY: TEST # 2536059856

PROBLEM	ANSWER	REMAINDER
1	3	
2	3	
3	1	
4	1	
5	3	

IF FINISHED, ENTER 0 ELSE ENTER 1
:

APPENDIX C

APPENDIX C

List of Symbols and Notation Used

a, b	Indexing subscript for persons.
α	Cronbach reliability coefficient | Type I error probability.
β	Type II error probability.
c	Error criterion.
C_{ak}	Natural proficiency limit.
d	(Appendix E) number of defects.
e	Natural logarithm base.
$f(x_a)$	Probability distribution function (binomial).
g, h, i, j	Indexing subscript for items.
k	Indexing subscript for tests, objectives.
L	Likelihood function.
m	Index for item responses.
n	Test length, in number of items.
p	Item-difficulty.
ϕ	Instructional Effectiveness Index.
q	1 - p
$R(\zeta' \mid S)$	Risk function, conditioned on sampling plan S.
S	Success probability; Sampling Plan label.
t	Time; number of trials.
$u_{ga} = 1, 0$	Item response observed for pupil a or item g.
U	Item response: random variable.
$\underset{\sim}{v}_{ak} = (u_{1a}, u_{2a}, \ldots, u_{ga})$	Item response pattern observed for pupil a on test k.
$\underset{\sim}{V}$	Item response pattern: random variable.

w	Number of error responses.
x	Number of correct responses observed.
$x_a(\underset{\sim}{v}) = \sum_{g=1}^{n} u_{ga}$	Scoring formula; test score observed.
X	Test score: random variable.
ζ	Proficiency.
ζ'	$1 - \zeta$; error rate.
ζ_1'	Nominal mastery limit (AQL).
ζ_2'	Nominal nonmastery limit (LTPD).
AQL	Average Quality Limit.
ASN	Average Sample Number.
CRT	Criterion-referenced Test.
CSSP	Curtailed Single Sampling Plan.
LTPD	Lot Tolerance Percent Defective.
OC	Operating Characteristic.
SCO	Specified Content Objective.
SPRT	Sequential Probability Ratio Test.
SSP	Single Sampling Plan.

APPENDIX D

APPENDIX D

Listings of two programs are included in Appendix D which are useful aids in writing acceptance requirements for the CSSP and SPRT.

1. GASNOCS Graph of Average Sample Numbers and Operating Characteristics.

GASNOCS solves equations (3-11) and (3-12), the OC and ASN for a CSSP. The independent variable is error rate. Values are arbitrarily selected in steps of .02 in the range from 0 to 1.00 inclusive. For each value of error rate, a corresponding dependent value is calculated for the OC and the ASN. These values are stored in the arrays named OC and ASN, one set for each of the criterion values (up to a maximum of ten) specified on the second parameter card. When data is accumulated for all criterion values associated with a given test length, graphs are drawn overlaid on one set of axes.

Multiple runs are made possible simply by adding pairs of parameter cards as described in the program documentation. The program stops when a parameter value zero is entered for test length.

2. Program KRI computes the ASN and OC for the SPRT. Unlike GASNOCS, curves are not overlaid. The program begins with the reading of four parameter values: α, β, ζ'_1 and ζ'_2. These values are entered on a parameter card divided into four 5-column fields (4F5.3).

The independent variable is a dummy quantity named Y in Program KRI. This is the same quantity as that called "x" in SRG (1945), p 2.50. Computational complexities lead one to employ the dummy variable and to select its values either by trial and error, which turns out to be inefficient, or by using an involved, but systematic technique recommended by the SRG authors. The latter was found to work very effectively. Recommended procedures found in SRG tend to favor approximate methods suitable for hand calculation inasmuch as high speed computers were not generally available at that time. The more formidable appearing formulas for exact calculation were found to be most efficient for CALCOMP graphing procedures.

As many parameter cards as desired may be included in the data deck. One set of OC and ASN curves is drawn for each card. Program terminates by sensing end-of-file (no more data cards to be read).

N.B.: (FORMAT 500 and the subsequent PRINT are useful only for crosschecking certain hand-calculations and may be removed without impairing the program.)

PROGRAM GASNOCS

```
C       FORTRAN 63 SOURCE DECK.
C       GASNOCS DRAWS ASN AND OC CURVES ON 8X8 GRAPHS.
C       TWO PARAMETER CARDS ARE NEEDED FOR EACH GRAPH.
C       THE FIRST CARD ENTERS THE TEST SIZE (N) AND NUMBER OF CRITERIA.
C       THE SECOND CARD CONTAINS THE ERROR CRITERIA VALUES.
C       ALL READ-FIELDS ARE I2.
C       SAMPLE PARAMETER CARDS ARE LISTED AT END OF PROGRAM.
C       A MAXIMUM O 10 OVERLAYS OR CRITERION VALUES IS ALLOWED.
C       PROGRAM TERMINATES BY READING A ZERO FOR N.
        DIMENSION ASN(51,10),OC(51,10),Y(51),Z(51),IC(10)
        TYPE INTEGER C
C       FUNCTION BF COMPUTES THE BINOMIAL COEFFICIENT
        BF(N,K)=FACT(N)/(FACT(K)*FACT(N-K))
    2   READ 1000,N,ML
        IF (N.EQ.0)50,3
   50   STOP
    3   READ 1000,(IC(I),I=1,ML)
        DO 4 I=1,51
        DO 4 J=1,10
        ASN(I,J)=0
    4   OC(I,J)=0
        DO 5 I=1,51
        Y(I)=0
    5   Z(I)=0
        IF (ML.GT.10)6,7
    6   ML=10
    7   DO 97 K=1,ML
        MX=2
        C=IC(K)
        L1=N-C+1
        L5=C-1
        DO 96 NX=1,49
        FX=NX
        X=(2*FX)/100.
        E=0$E1=0$E2=0$PA=0
C       LOOP 110 CALCULATES PROBABILITY OF TEST ENDING IN NONACCEPTANCE,
C         THE NO. OF ITEMS SAMPLED AT TIME OF CURTAILMENT, THE EXPECTED
C         TEST LENGTH FOR TERMINATION IN REJECTION, AND PART OF THE DIST.
        DO 110 M=1,L1
        L4=C+M-2$L8=C+M-1$G8=L8
        PNR=BF(L4,L5)*X**C*(1.-X)**(M-1)
        E1=E1+PNR*G8
  110   CONTINUE
C       LOOP 120 CALCULATES QUANTITIES AS IN 110 ONLY FOR TESTS ENDING IN
C         A DECISION TO ACCEPT (I.E. ATTAINMENT OF MASTERY IS INFERRED).
        DO 120 M=1,C
        L6=N-C+M-1$FC=N-C+M$L2=FC$L7=M-1
        PNA=BF(L6,L7)*X**(M-1)*(1.-X)**L1
        PA=PA+PNA
  120   E2=E2+PNA*FC
        E=E1+E2
        ASN(MX,K)=E
        OC(MX,K)=PA
        MX=MX+1
```

```
C       PROGRAM GASNOCS (CONTINUED)
   96 CONTINUE
   97 CONTINUE
C       SET UP 1-DIMENSIONAL ARRAYS BEFORE CALLING GRAPH SEQUENCE FOR
C         OC'S AND ASN'S.
       MX=1
       DO 200 I = 1,101,2
       FI=I
       Z(MX)=(FI-1.)/100.
  200  MX=MX+1
       Y(51)=0.
       Y(1)=1.
       FN=N
       DO 94 K=1,ML
       DO 93 MX=2,50
    93 Y(MX)=OC(MX,K)
       IF (K.EQ.1)931,932
   931 CALL GRAPH1(Z,Y,51,3H8X8,4HAUTO,
      1 48H      OC CURVES FOR FIXED N AND SELECTED CRITERIA,
      2 16H     ERROR RATE..,21HSUCCESS PROBABILITY..)
       GO TO 94
   932 CALL GRAPH1(Z,Y,51,7HOVERLAY,4HSAME)
    94 CONTINUE
       DO 92 K=1,ML
       C=IC(K)
       Y(1)=N-C+1
       Y(51)=C
       DO 91 MX=2,50
    91 Y(MX)=ASN(MX,K)
       IF (K.EQ.1)911,912
   911 CALL GRAPH1(Z,Y,51,3H8X8,4HAUTO,
      1 48H     ASN CURVES FOR FIXED N AND SELECTED CRITERIA,
      2 16H     ERROR RATE..,23HAVERAGE SAMPLE NUMBER..)
       GO TO 92
   912 CALL GRAPH1(Z,Y,51,7HOVERLAY,4HSAME)
       GO TO 2
  1000FORMAT (40I2)
       END

       FUNCTION FACT(M)
       FACT=1.
       DO 5 I=1,M
       FI=I
     5 FACT=FACT*FI
       RETURN
       END
       END
          FINIS
'EXECUTE.
0502
0102
2008
0102030405060708
2510
01020304050607080910
0000
```

```
      PROGRAM KRI
    DIMENSION XPU(42),SSU(42),NBA(42)
    TYPE REAL NBA
  3 READ 100,ALP,BET,XUP,XZP
100 FORMAT(4F5.3)
    IF(EOF,50)99,4
  4 ONE=XZP/XUP
    TWO=(1-XUP)/(1-XZP)
    THREE=(1-BET)/ALP
    FOUR=(1-ALP)/BET
    GU=ALOG10(ONE)
    GD=ALOG10(TWO)
    A=ALOG10(THREE)
    B=ALOG10(FOUR)
    HU=B/(GU+GD)
    HD=A/(GU+GD)
    HB=HU+HD
    S=GD/(GU+GD)
500 FORMAT(8F14.5)
    PRINT 500,GU,GD,A,B,HU,HD,HB,S
    BB=.1*(GU+GD)
    CON=POWRF(2.7182818284,BB)
    YEXP=1.
    ADD=0.
    DO 1 N=1,20
    YEXP=YEXP+ADD
    Y=CON**YEXP
    XPU(22-N)=(Y**S-1)/(Y-1)
    XPU(21+N)=XPU(22-N)*(Y**(1-S))
    SSU(22-N)=((Y**HB)-(Y**HU))/(Y**HB-1)
    SSU(21+N)=SSU(22-N)/(Y**HU)
  1 ADD=ADD+1.
    XPU(1)=0.
    XPU(42)=1.
    SSU(1)=1.
    SSU(42)=0.
    DO 2 N=1,42
  2 NBA(N)=(SSU(N)*HB-HD)/(S-XPU(N))
    CALL GRAPH1(XPU,SSU,42,4H8X8,4HAUTO,
   141HASN CURVES FOR SEQUENTIAL RATIO TESTING..,
   212HERROR RATE
   323HAVERAGE SAMPLE NUMBER..)
    GO TO 3
 99 END
```

APPENDIX E

APPENDIX E

Notes on the Sequential Probability Ratio Test
(adapted from SRG - 1945)

1. A sequential sampling plan is defined by the specification of four quantities: α, β, ζ'_1, and ζ'_2 where

   ζ'_1 = limit of the region of acceptance for H_0 (nominal mastery point.)

   ζ'_2 = limit of the region of acceptance for H_1 (nominal nonmastery point.)

   α = probability of accepting H_1 when $\zeta' = \zeta'_1$

   β = probability of accepting H_0 when $\zeta' = \zeta'_2$

2. The hypotheses H_0 and H_1 for the two decision case are simply

   H_0: The student has attained a mastery level of proficiency

   H_1: The student has not attained a mastery level of proficiency

3. Decision rules are conveniently used in the following form (adapted from Wald (1947)):

   (a) Let w = number of errors in the first m items tested.

   ζ' = pupil error rate

(1) Continue inspection as long as

$$\log\left(\frac{\beta}{1-\alpha}\right) < \log\left(\frac{\zeta'_2{}^{w}\,(1-\zeta'_2)^{m-w}}{\zeta'_1{}^{w}\,(1-\zeta'_1)^{m-w}}\right) < \log\left(\frac{1-\beta}{\alpha}\right)$$

(2) Accept H_1 if on the last item tested

$$\log\left(\frac{\zeta'_2{}^{w}\,(1-\zeta'_2)^{m-w}}{\zeta'_1{}^{w}\,(1-\zeta'_1)^{m-w}}\right) \geq \log\left(\frac{1-\beta}{\alpha}\right)$$

(3) Accept H_0 if on the last item tested

$$\log\left(\frac{\zeta'_2{}^{w}\,(1-\zeta'_2)^{m-w}}{\zeta'_1{}^{w}\,(1-\zeta'_1)^{m-w}}\right) \leq \log\left(\frac{\beta}{1-\alpha}\right)$$

4. A selected plan will accept H_1 with frequency $\leq \alpha$ when $\zeta' \leq \zeta'_1$ (Type I error); the plan will accept H_0 with frequency $\leq \beta$ when $\zeta' \geq \zeta'_2$ (Type II error). If $\zeta'_2 > \zeta' > \zeta'_1$ the plan will accept H_0 with relative frequency between $(1-\alpha)$ and β, the frequency decreasing progressively from $(1-\alpha)$ to β as ζ' ranges from ζ'_1 to ζ'_2.

5. The description of sampling plans in (4) is typical of both the SPRT and CSSP discussed in this paper. The sequential sampling

plan based on the SPRT can be used to accept or not accept samples with substantially fewer questions than other available plans under the conditions stated in part 7, below.

6. Construction of the lattice diagram for the SPRT sampling plan may be accomplished by using the following relations:

   a. Let the horizontal axis be scaled in terms of the total number of questions asked; label the axis n.

   b. Let the vertical axis be scaled in terms of the number of erroneous responses detected; the vertical axis is labeled d.

   c. The limit lines which signal end of test with decision for H_0 or H_1 are given by

   (1) $d_1 = -h_1 + sn$ (lower limit line)

   (2) $d_2 = h_2 + sn$ (upper limit line)

   where d = number of errors

   n = number of test items

   s = slope of line

   h_i = vertical axis intercepts; i = 1, 2.

   d. Given α, β, ζ'_1, and ζ'_2, the lines can be determined by computing slopes and intercepts from the following formulas:

$$(1) \quad h_1 = \frac{\log \left(\frac{1 - \alpha}{\beta}\right)}{\log \left(\frac{\zeta'_2 (1 - \zeta'_1)}{\zeta'_1 (1 - \zeta'_2)}\right)}$$

$$(2)\quad h_2 = \frac{\log\left(\frac{1-\beta}{\alpha}\right)}{\log\left(\frac{\zeta'_2\,(1-\zeta'_1)}{\zeta'_1\,(1-\zeta'_2)}\right)}$$

$$(3)\quad s = \frac{\log\left(\frac{1-\zeta'_1}{1-\zeta'_2}\right)}{\log\left(\frac{\zeta'_2}{\zeta'_1}\cdot\frac{(1-\zeta'_1)}{(1-\zeta'_2)}\right)}$$

e. Calculation is most easily effected by first computing

$$(1)\quad g_1 = \log\left(\frac{\zeta'_2}{\zeta'_1}\right)$$

$$(2)\quad g_2 = \log\left(\frac{1-\zeta'_1}{1-\zeta'_2}\right)$$

$$(3)\quad a = \log\left(\frac{1-\beta}{\alpha}\right)$$

$$(4)\quad b = \log\left(\frac{1-\alpha}{\beta}\right)$$

Then

$$(5)\quad h_1 = \frac{b}{g_1 + g_2}$$

(6) $h_2 = \frac{a}{g_1 + g_2}$

(7) $s = \frac{g_2}{g_1 + g_2}$

f. As a check against the accuracy of computation, note that

(1) $\zeta'_1 < \zeta'_2$

(2) $\alpha + \beta < 1$

(3) $\zeta'_1 < s < \zeta'_2$

(4) $h_1 = h_2$ if $\alpha = \beta$

7. The SPRT is an optimum sampling plan if $\zeta' = \zeta'_1$ or if $\zeta' = \zeta'_2$. For these particular values of proficiency, the Average Sample Number (ASN) required to reach a decision is calculated as follows:

(a) If $\zeta' = \zeta'_1$, $\bar{n}_{\zeta'_1} = \frac{(1 - \alpha)h_1 - \zeta h_2}{s - \zeta'_1}$

(b) $\bar{n}_{\zeta'_2} = \frac{(1 - \beta)h_2 - \beta h_1}{\zeta'_2 - s}$

8. The OC-curve for the SPRT is obtained in the following manner:

a. Let S = probability of accepting H_0 (as a function of ζ')

(Note S = "success" probability).

The Oc-curve is plotted with S as the vertical axis

and ζ' as the horizontal axis where

$$0 \leq S \leq 1$$

and $$0 \leq \zeta' \leq 1$$

b. The simplest approach is to locate the following five points on the OC-curve and interpolate between them:

(1) when $\zeta' = 0$; $S = 1$

(2) when $\zeta' = \zeta'_1$, $S = 1 - \alpha$

(3) when $\zeta' = s$, $S = \dfrac{h_2}{h_1 + h_2}$

(4) when $\zeta' = \zeta'_2$, $S = \beta$

(5) when $\zeta' = 1$, $S = 0$

9. ASN for the SPRT is computed as follows:

(a) $$ASN = \bar{n}_{\zeta'} = \frac{S.(h_1 + h_2) - h_2}{s - p}$$

Note: Select arbitrary values of $S(\zeta')$ in the range $0 \leq \zeta' \leq 1$.

Compute and plot $(\zeta', \bar{n}_{\zeta'})$

10. Termination formula:

$$\text{If } w_m \log \frac{\zeta'_2 (1 - \zeta'_1)}{\zeta'_1 (1 - \zeta'_2)} - m \log \left(\frac{1 - \zeta'_1}{1 - \zeta'_2}\right) \begin{cases} > 0 \text{ accept } H_1 \\ < 0 \text{ accept } H_0 \end{cases}$$

APPENDIX F

APPENDIX F

The following pages contain a listing of program and data files used to demonstrate certain aspects of a computer managed system of mathematics instruction.

The first document is a listing of the ALGOL source program called MATH/CMS as described in Chapter Four. This program was designed as a model which illustrates in a simple way the basic concepts discussed in this dissertation.

The parametric specification for twenty SCO's is shown in the listing of the INPUT File STAN/CMS. MATH/CMS accesses these parameters when a non-zero test number is requested.

Anyone of the three kinds of sampling plans discussed in Chapter Three may be implemented by simply entering the values 0, 1, or 2 (for SSP, CSSP, and SPRT) in STAN/CMS under the column headed by "S", (parameter 20). If the CSSP is selected, a two-digit error criterion should be entered under the column headed by "CR" (parameter 21). If the SPRT is desired, one must specify the values for Z1, Z2, AL, and BT (or ζ'_1, ζ'_2, α, and β). The program MATH/CMS will then implement the selected sampling procedure when operating in interactive modes one or four.

The SSP is automatically selected by the program MATH/CMS in Mode 2,

since this is a testing mode which forms homogeneous proficiency groups using test score variance limits rather than fixed criterion limits. The technique requires that all pupils respond to the same items and that the number of items be the same for all pupils. Thus Mode 2 selects a fixed sampling plan and, after random generation of the first set of items, resets the random number generator automatically to regenerate the same sample for subsequent examinees. The output from Mode 2 is written on the file ITEMS/CMS in the form of item response patterns, v_{ak}. The pupil also receives a summary of his error rate and latency upon completion of the test. The file ITEMS/CMS then becomes the input file for the program which sorts the examinees into homogeneous proficiency groups.

Output from all interactive modes is kept on a student history file called GIESE/CMS, GIESE being the name of the school where the data is generated. A brief segment of the data maintained on this file is shown at the end of this appendix. Two files not listed here are called GIESEN/CMS and GIESEM/CMS (for nonmaster and master, respectively). The data format of each is identical to that of GIESE/CMS. They are simply used to sort pupils into mastery and nonmastery groups when using modes one or four with either the CSSP or SPRT. This provides the fastest turn-around time when grouping information is desirable and obtainable through interactive testing. Experience to date indicates this service will have usefulness both in the primary grades and at the intermediate levels where regular skill maintenance programs may be needed.

Present operation protocols use the computer management system in the following manner. The basic system, called = /CMS, containing the main program and initialized files are written on duplicate tapes (U7358, U7359). At the beginning of the school day, the aide loads = /CMS (from tape to disk) to prepare for operation. During the day, tests and practice sheets may be obtained (Mode 0) and administered in the conventional noninteractive manner. The aide scores the papers using a key provided by the computer at the time of test generation. Item response patterns may be entered in File ITEMS/CMS using the systems program EDIT/EDIT, if the purpose is to form homogeneous proficiency groups. This would usually be the case when pre-test data is obtained. In other cases, the pupil's performance is simply entered on the history file GIESE/CMS, to maintain the record of his performance.

In cases where mastery or nonmastery decisions are made, history tape files will be maintained with data updated from GIESEN/CMS, GIESEM/CMS, or according to decision rules applied by the aide when hand scoring is used.

In this way an overall performance file is maintained on tape for all pupils; corresponding history tape files are maintained for masters and nonmasters of specified content objectives; and the size of disk data files is kept to a minimum by beginning each day with initially empty daily data files.

At intervals, it will be required to gather information regarding the overall performance of individual pupils. If these records are too long for output on the teletype, the aide will use EDIT to create

individual pupil performance records for specified intervals of time from the several tape-sources. These pupil-files can be printed on remote command on the line printer at the UWCC. Facilities exist for mailing reports of this kind to the school on the same day the report is printed. This service should be most useful in advising pupils, prescribing short term programs of mathematics study, and preparing regular reports to parents.

The last four pages in Appendix F show test samples of interactive exchanges in which various functions of the system are exercised. The first is a delimited 10-item random sample of the four fundamental operations on whole numbers together with a random sample of horizontal and vertical formats. The second and third samples show the appearance of interactive sequences terminated with mastery or nonmastery decisions.

The final sample shows illustrative daily disk files kept in the several kinds of records described earlier. Normally, these daily files would be copied on the end of the history tape-file prior to erasing the disk of all CMS files at the end of the school day.

LABEL OMATHC4SOLDOUBLE 601321492 RUN EDIT 5/29 10:31:0

MATH /CMS LISTED AT 10:59 ON 69149 BY

```
COMMENT--SOURCE PROGRAM MATH/CMS;
BEGIN                                                                    00000200
  LABEL STRTR;                                                           00000300
  DEFINE DCINCHAR=28#,NBUFIN=3#,BUFINSIZE=5#,DCOUTCHAR=27#,NBUFOUT=5#,   00000400
   BUFOUTSIZE=5#,MSGSIZE=80#,ANSSIZE=120#,WAITTIME=300#,WRITEWATT=150#;  00000500
   ARRAY MESSAGE[0:(MSGSIZE-1)DIV 8],ANSWER[0:(ANSSIZE-1)DIV 8];         00000600
  ALPHA                                                                  00000700
  FILE IN DCIN 14 (NBUFIN,BUFINSIZE);                                    00000800
  ALPHA                                                                  00000900
  FILE OUT DCOUT 14 (NBUFOUT,BUFOUTSIZE);                                00001000
  REAL USER;                                                             00001100
  BOOLEAN INUSE,BREAK;                                                   00001200
  BOOLEAN                                                                00001300
  PROCEDURE OUTPUT;                                                      00001400
   BEGIN                                                                 00001500
     LABEL LOOP,NOGO,BROKE,EXIT;                                         00001600
     INTEGER CNT;                                                        00001700
     BOOLEAN GPMK;                                                       00001800
     REAL S;ARRAY TANK[0:BUFOUTSIZE-1];                                  00001900
     STREAM                                                              00002000
     PROCEDURE STOPPER(A);                                               00002100
      BEGIN                                                              00002200
        DI←A;DI←DI+4;DS←3 LIT "≤≠←"                                      00002300
      END;                                                               00002400
     BOOLEAN                                                             00002500
     STREAM                                                              00002600
     PROCEDURE MOVE(A,B,C);VALUE C;                                      00002700
      BEGIN                                                              00002800
        SI←A;SI←SI+C;DI←B;TALLY←0;DCOUTCHAR(                             00002900
        IF SC="←"THEN TALLY←1;DS←CHR);DS←LIT "←";MOVE←TALLY;             00003000
      END MOVE;                                                          00003100
      IF NOT INUSE THEN                                                  00003200
      BEGIN                                                              00003300
        IF STATUS(TANK[*])≠2 THEN GO TO EXIT;USER←TANK[0];INUSE←TRUE;    00003400
        SEEK(DCIN(USER));                                                00003500
      END;STOPPER(ANSWER[(ANSSIZE-1)DIV 8]);OUTPUT←TRUE;TANK[0]←USER;    00003600
      LOOP:WHILE TRUE DO                                                 00003700
      BEGIN                                                              00003800
        GPMK←MOVE(ANSWER[CNT DIV 8],TANK[1],ENTIER(CNT MOD 8)):WRITE (   00003900
        DCOUT(USER,WRITEWATT),BUFOUTSIZE ,TANK[*])[NOGO:BROKE];          00004000
        IF GPMK THEN GO TO EXIT;CNT←CNT+DCOUTCHAR;                       00004100
```

```
    END;BROKE:BREAK←TRUE;GO TO EXIT;NOGO:CLOSE(DCIN);CLOSE(DCOUT);      00004200
    OUTPUT←INUSE←FALSE;USER←0;EXIT:                                     00004300
 END OUTPUT;                                                            00004400
BOOLEAN                                                                 00004500
PROCEDURE INPUT;                                                        00004600
 BEGIN                                                                  00004700
    SWITCH FORMAT FLAGS←("≤≠←"),(                                       00004800
    "≤≠  MESSAGE WAS TOO LONG...PLEASE REENTER≤≠←"),(                   00004900
    "≤≤  MESSAGE WAS PARTLY LOST...PLEASE REENTER≤≠←")];ARRAY SINK[0:    00005000
    BUFINSIZE-1];                                                       00005100
   INTEGER CNT,TIMEX,FLAG,Q,R,N;                                        00005200
   LABEL LOOP,NOGO,ABNORM,CHECKIT,EXIT,BADNEWS;                         00005300
   REAL S;                                                              00005400
   INTEGER                                                              00005500
   STREAM                                                               00005600
   PROCEDURE SIZE(A);                                                   00005700
    BEGIN                                                               00005800
       SI←A;TALLY←1;DCINCHAR(                                           00005900
       IF SC="←"THEN JUMP OUT;SI←SI+1;TALLY←TALLY+1));SIZE←TALLY        00006000
    END SIZE;                                                           00006100
   STREAM                                                               00006200
   PROCEDURE MOVE(A,B,C,D);VALUE C,D;                                   00006300
    BEGIN                                                               00006400
       SI←A;DI←B;DT←DT+D;DS←C CHR                                       00006500
    END MOVE;                                                           00006600
    IF NOT INUSE THEN                                                   00006700
    BEGIN                                                               00006800
       IF STATUS(SINK[*])≠2 THEN GO TO EXIT;USER←SINK[0];INUSE:=TRUE;   00006900
       SEEK(DCIN(USER));                                                00007000
    END LOOKING FOR A USER;N←DCINCHAR;BREAK←FALSE;LOOP:DO               00007100
    BEGIN                                                               00007200
       READ(DCIN(USER,WAITTIME),BUFINSIZE,SINK[*1)[NOGO:ABNORM];        00007300
       IF SINK[0].[25:1]=0 THEN N←SIZE(SINK[1]);                        00007400
       IF CNT+N GTR MSGSIZE THEN FLAG←1                                 00007500
       ELSE MOVE(SINK[1],MESSAGE[CNT DIV 8],N,ENTIER(CNT MOD 8));CNT←   00007600
       CNT←N;                                                           00007700
    END UNTIL SINK[0].[25:1]=0;NOGO:CHECKIT:WRITE(ANSWER[*],FLAGS[FLAG  00007800
    ]);                                                                 00007900
    IF NOT OUTPUT THEN GO TO BADNEWS;                                   00008000
    IF FLAG≠0 THEN                                                      00008100
    IF INPUT THEN GO TO EXIT                                            00008200
```

```
    ELSE GO TO BADNEWS;GO TO EXIT;ABNORM;                                    00008300
    IF SINK[0].L28:1]=1 THEN                                                 00008400
    BEGIN                                                                    00008500
       FLAG←2;GO TO CHECKIT;                                                 00008600
    END;BADNEWS:CLOSE(DCIN);CLOSE(DCOUT);USER←0;INPUT←INUSE←FALSE;EXIT       00008700
    ;                                                                        00008800
 END INPUT;                                                                  00008900
INTEGER                                                                      00009000
PROCEDURE RANDOM(MAX,MIN,START);                                             00009100
REAL MAX,MIN,START;                                                          00009200
 BEGIN                                                                       00009300
   DEFINE BASE =34359738358#;OWN                                             00009400
   REAL PREVIOUS;                                                            00009500
    IF START≠0 THEN PREVIOUS←START;PREVIOUS←4357×PREVIOUS MOD BASE;          00009600
    RANDOM←(PREVIOUS/BASE×(MAX-MIN+1))DIV 1 +MIN                             00009700
 END                                                                         00009800
PROCEDURE RANDOM;                                                            00009900
PROCEDURE ZEROCK(N,A,B,DIVR,DIVD,REM,BS);                                    00010000
INTEGER N,A,B,DIVR,DIVD,REM,BS;                                              00010100
 BEGIN                                                                       00010200
   INTEGER I,K,L;                                                            00010300
   REAL AP1,AP2,AP3;AP1←DIVR-1;AP2←1;AP3←0;K←0;                              00010400
    IF B ≠A THEN L ←(B-A)                                                    00010500
    ELSE L ←1;FOR I←A STEP L UNTIL B DO N←N-((N MOD BS*I)DIV BS*(I-1))       00010600
    ×BS*(I-1);                                                               00010700
    IF REM ≥0 THEN DIVD ←DIVR×N +REM×RANDOM(AP1,AP2,K)                       00010800
    ELSE DIVD ←DIVR×N +RANDOM(AP2,AP3,K)×RANDOM(AP1,AP2,K);                  00010900
 END ZEROCK;                                                                 00011000
COMMENT--ZEROC< GENERATES A DIVISION PROBLEM WITH ZEROES IN THE ATH AND      00011100
BTH PLACES OF THE QUOTIENT;                                                  00011200
 PROCEDURE CARYCOV(OP1,OP2,DIG,OPN,RGP,DIGM,REM,ANS,BS);                     00011300
INTEGER OP1,OP2,DIG,OPN,RGP,DIGM,REM,ANS,BS;                                 00011400
 BEGIN                                                                       00011500
   REAL AP1,AP2,AP3;                                                         00011600
   INTEGER I,J,A,B;                                                          00011700
   LABEL ENDI,AD,SB,MT,ML,DV,N2,LOOP,GOUT;SWITCH SELCT←AD,SB,MT,DV;J←        00011800
    ;                                                                        00011900
    IF OPN =3 THEN GO TO DV;LOOP:FOR I←1 STEP 1 UNTIL DIG DO                 00012000
    BEGIN                                                                    00012100
       A ←(OP1 MOD BS*I)DIV BS*(I-1);                                        00012200
       IF OPN =2 THEN GO TO ML;J ←I;N2:B ←(OP2 MOD BS *J)DIV BS*(J-1);       00012300
```

```
      GO TO SELCTIOPN+11;AD:                                                    00012400
      BEGIN                                                                     00012500
COMMENT--BLOCK C1;                                                              00012600
        INTEGER C;C ←(A+B)-(BS-1);                                              00012700
         IF C >0 AND RGP =0 THEN                                                00012800
         BEGIN                                                                  00012900
            IF B =(BS-1)THEN                                                    00013000
            BEGIN                                                               00013100
               OP2 ←OP2-B×BS*(I-1);B ←RANDOM ((BS-2),0,0);OP2 ←OP2 +B00013200
               ×BS*(I-1);                                                       00013300
            END;OP1←OP1 -A×BS*(I-1);AP1 ←(BS-1)-B;                             00013400
            IF I ≠DIG THEN A ←RANDOM (AP1,0,0)                                 00013500
            ELSE A ←RANDOM(AP1,1,0);OP1 ←OP1 +A×BS*(I-1);GO TO ENDI;           00013600
         END;                                                                   00013700
         IF C ≤0 AND RGP =1 THEN                                                00013800
         BEGIN                                                                  00013900
            OP1 ←OP1 -A×BS*(I-1);OP2 ←OP2 -B×BS*(I-1);A ←RANDOM ((BS-00014000
            1),1,0);AP1 ←BS -A;B ←RANDOM ((BS-1),AP1,0);OP1 ←OP1 +A× 00014100
            BS*(I-1);OP2 ←OP2 +B×BS*(I-1);GO TO ENDI;                           00014200
         END;GO TO ENDI;                                                        00014300
       END BLOCK C1;SB:                                                         00014400
       BEGIN                                                                    00014500
COMMENT--BLOCK C2;                                                              00014600
          INTEGER C;C ←A -B;                                                    00014700
           IF C<0 AND RGP =0 THEN                                               00014800
           BEGIN                                                                00014900
              OP1 ←OP1 -A×BS*(I-1);OP2 ←OP2 -B×BS*(I-1);                        00015000
              IF I ≠DIG THEN A ←RANDOM((BS-1),0,0)                              00015100
              ELSE A ←RANDOM ((BS-1),1,0);                                      00015200
              IF I ≠DIG THEN B ←RANDOM(A,0,0)                                   00015300
              ELSE B ←RANDOM (A,1,0);OP1 ←OP1 +A×BS*(I-1);OP2 ←OP2 +B× 00015400
              BS*(I-1);GO TO ENDI;                                              00015500
           END;                                                                 00015600
           IF C ≥0 AND RGP =1 THEN                                              00015700
           BEGIN                                                                00015800
              OP1 ←OP1 -A×BS*(I-1);OP2 ←OP2 -B×BS*(I-1);B ←RANDOM ((BS-00015900
              1),1,0);AP1 ←B-1;                                                 00016000
              IF I ≠DIG THEN A ←RANDOM(AP1,0,0)                                 00016100
              ELSE A ←RANDOM((BS-1),B,0);OP1 ←OP1 +A×BS*(I-1);OP2 ←OP2 00016200
              +B×BS*(I-1);GO TO ENDI;                                           00016300
           END REGROUP;GO TO ENDI;                                              00016400
```

```
        END BLOCK C2;ML:                                                    00016500
        IF I =1 THEN                                                        00016600
        BEGIN                                                               00016700
            J ←J +1;GO TO N2;                                               00016800
        END;MT:                                                             00016900
        BEGIN                                                               00017000
COMMENT--BLOCK C3;                                                          00017100
          INTEGER C;C ←(A×B)-(BS-1);                                        00017200
           IF C >0 AND RGP =0 THEN                                          00017300
           BEGIN                                                            00017400
              IF A >4 THEN                                                  00017500
              IF RANDOM(1,0,0)=0 THEN                                       00017600
              BEGIN                                                         00017700
                 OP2 ←OP2 -B×BS*(J-1);B ←RANDOM(4,1,0);OP2 ←OP2 +B×BS*(   00017800
                 J-1);                                                      00017900
              END;OP1←OP1 -A×BS*(I-1);AP1 ←(BS-1)DIV B;                     00018000
              IF I =DIG OR I =1 THEN A ←RANDOM (AP1,1,0)                    00018100
              ELSE A ←RANDOM(AP1,0,0);OP1 ←OP1 +A×BS*(I-1);GO TO ENDI;      00018200
           END;                                                             00018300
           IF C ≤0 AND RGP =1 THEN                                          00018400
           BEGIN                                                            00018500
              OP1 ←OP1 -A×BS*(I-1);OP2 ←OP2 -B×BS*(J-1);                    00018600
              IF B <2 THEN                                                  00018700
              IF BS >2 THEN B ←RANDOM ((BS-1),2,0)                          00018800
              ELSE B←RANDOM(BS,1,0)                                         00018900
              ELSE B←RANDOM((BS-1),B,0);AP1 ←((BS-1)DIV B)+1;A ←RANDOM      00019000
              ((BS-1),AP1,0);OP1 ←OP1 +A×BS*(I-1);OP2 ←OP2 +B×BS*(J-1);     00019100
              GO TO ENDI;                                                   00019200
           END;GO TO ENDI;                                                  00019300
        END BLOCK C3;ENDI:                                                  00019400
    END OF ILOOP:                                                           00019500
    IF OPN ≠2 THEN J ←DIGM +1;                                              00019600
    IF J <DIGM THEN GO TO LOOP                                              00019700
    ELSE GO TO GOUT;DV:                                                     00019800
    BEGIN                                                                   00019900
COMMENT--BLOCK C4;                                                          00020000
      INTEGER C;                                                            00020100
      LABEL ENDJ;                                                           00020200
       IF OP1 MOD OP2 ≠0 AND DIG =0 AND REM =0 THEN OP1 ←OP1 -(OP1 MOD      00020300
        OP2)+OP2×RANDOM(2,1,0);                                             00020400
       IF OP1 MOD OP2 =0 AND DIG =0 AND REM >0 THEN                         00020500
```

```
      BEGIN                                                          00020600
         AP1 ←OP2-1;OP1 ←OP1 +RANDOM(AP1,1,0);                       00020700
      END;                                                           00020800
      IF REM <0 THEN GO TO GOUT;                                     00020900
      IF OP2 =0 THEN OP2←RANDOM((BS-1),2,0);B ←OP2;FOR J ←DIG STEP -1 00021000
       UNTIL 1 DO                                                    00021100
      BEGIN                                                          00021200
COMMENT--J LOOP;A←(OP1 MOD (BS*J))DIV BS*(J-1);C ←A MOD B;           00021300
         IF C ≠0 AND BGP =0 THEN                                     00021400
         BEGIN                                                       00021500
            OP1 ←OP1-A×BS*(J-1);AP1 ←(BS-1)DIV B;                    00021600
            IF J=DIG THEN A ←A-C+B×RANDOM(AP1,0,0)                   00021700
            ELSE A ←A-C;OP1 ←OP1 +A×BS*(J-1);GO TO ENDJ;             00021800
         END;                                                        00021900
         IF C =0 AND BGP =1 THEN                                     00022000
         BEGIN                                                       00022100
            AP1 ←B-1;A ←A +RANDOM (AP1,1,0);OP1←OP1 +A×BS*(J-1);     00022200
         END;ENDJ:                                                   00022300
      END OF J LOOP;                                                 00022400
   END BLOCK C4;GOUT:                                                00022500
 END CARYCON;                                                        00022600
PROCEDURE WOUT1(F,L);FORMAT F;LIST L;                                00022700
 BEGIN                                                               00022800
    WRITE(ANSWER[*1,F,L);                                            00022900
    IF OUTPUT THEN;                                                  00023000
 END WOUT1;                                                          00023100
PROCEDURE WOUT2(F);FORMAT F;                                         00023200
 BEGIN                                                               00023300
    WRITE(ANSWER[*1,F);                                              00023400
    IF OUTPUT THEN;                                                  00023500
 END WOUT2;                                                          00023600
PROCEDURE WOUT3(F,N,L);SWITCH FORMAT F;LIST L;                       00023700
INTEGER N;                                                           00023800
 BEGIN                                                               00023900
    WRITE(ANSWER[*1,F[N],L);                                         00024000
    IF OUTPUT THEN;                                                  00024100
 END WOUT3;                                                          00024200
PROCEDURE WOUT4(F,N);SWITCH FORMAT F;                                00024300
INTEGER N;                                                           00024400
 BEGIN                                                               00024500
    WRITE(ANSWER[*1,F[N]);                                           00024600
```

```
    IF OUTPUT THEN;                                                    0002470
  END WOUT4;                                                           0002480
 PROCEDURE RIN1(F,L);FORMAT F;LIST L;                                  0002490
  BEGIN                                                                0002500
    IF INPUT THEN;READ(MESSAGE[*],F,L);                                0002510
  END RIN1;                                                            0002520
 PROCEDURE RIN3(F,N,L);SWITCH FORMAT F;LIST L;                         0002530
  INTEGER N;                                                           0002540
  BEGIN                                                                0002550
    IF INPUT THEN;READ(MESSAGE[*],F[N],L);                             0002560
  END RIN3;STRTR:                                                      0002570
  BEGIN                                                                0002580
COMMENT--BLOCK 00;                                                     0002590
    FILE IN PARAM DISK RANDOM "STAN" "CMS"(1,10,150,SAVE 30);          0002600
         FILE ITEMSCR DISK SERIAL "ITEMS" "CMS"(1,10,150,SAVE 30);     0002610
    FILE MASTER DISK SERIAL "GIESEM" "CMS"(1,10,150,SAVE 30);          0002620
    FILE NONMAST DISK SERIAL "GIESEN" "CMS"(1,10,150,SAVE 30);         0002630
    REAL ARRAY LATBAR[0:1],TRESP[1:25],TIT[1:12],PT[0:3],STD[0:1,1:    0002640
     12],KEY[1:25,1:2],PAR[1:26],RPTR[1:25,0:5],OP[1:5];               0002650
    REAL A1,B,FLAG,LIMAST,LINNON,L,MFL,NPT,MPT,SVSTRT,REM,STRTVAL,PPL, 0002660
     RPT,J,SAVSTRT,TRYCOL,HVFL,TP,OPFLAG,MODE,TID,OPN,PTR,PT,MOD3FL,   0002670
     MOD2FL,VPTR,HORPTR,OUTPTR,HV,TEMP,RFLAG,TUD,STT,NUM,WRNG,I,A,OP1, 0002680
     OP2,ANS,K,II;                                                     0002690
    LABEL BEG,TESTPT,BL4,EOF;LIST Z3 (STD[1,1],STD[1,2]);LIST Z21 (    0002700
     RPT);LIST Z4 (OP[1],OP[2]);LIST Z50 (FOR I ←1 STEP 1 UNTIL 12 DO  0002710
     TIT[I]);LIST Z55 (TUD,STRTVAL);FORMAT NAME2(A6,A3,2A6,A3),N4(     0002720
     "TEST #",I3,I6,X8,"NAME≠-----------------------------≤≠←"),       0002730
     NAME5(X25,"DATE≠-----------------------------≤≠←"),NAME3("≤≠"     0002740
     ,2A6,A3,A6,A3,T8,3(X2,T2,X2),X2,I2,X1,A6,X2,T3,X1,F4.1 ,"←"),     0002750
     PARFOR(4I1,4I5,2I8,I2,T1,2I2,2I1,I2,3I1,6I2),TITLE("≤≠",12A6,"←"),0002760
     RTII(12A6),F1("≤≠≠←"),R1(I1),R2(2I2,I8,A6),R4(I4),R7(2I1),F71(    0002770
     "≤≠TO START A NEW MODE ENTER 1 ELSE ENTER 0≤≠:←");FOR I ←1 STEP 1 0002780
     UNTIL 26 DO PAR[I]←0;FOR I ←0 STEP 1 UNTIL 1 DO FOR J ←1 STEP 1   0002790
     UNTIL 5 DO STD[I,J]←"      ";TP←TID←MOD2FL←HVFL←MODE←TRYCOL←MOD3FL 0002800
     ←PPL←SAVSTRT←0;READ(MASTER[0],R4,MPT);READ(NONMAST[0],R4,NPT);    0002810
     READ(ITEMSCR[0],R4,PT);BEG:                                       0002820
     BEGIN                                                             0002830
COMMENT--BLOCK 01;                                                     0002840
       REAL N,PCK;                                                     0002850
       LABEL OVER,TESTID,MOD2,GOON,GOGO,SET2FL;FORMAT NAME("≤≠",X5,    0002860
        "NAME = FIRST----L--A--S--T-----≤≠",X12,"←"),F31(              0002870
        "≤≠LETS GO WITH ≤≠",12A6,"←"),F32(                             0002880
```

```
"≤≠SELECT A TEST AS FOLLOWS:←"),F14(                                    00028900
"≤≠ENTER 1 IF CORRECT ANSWERS ARE TO BE PRINTED",                       00029000
" ELSE ENTER ZERO.≤≠:←"),F9(                                            00029100
"≤≠TYPE IN THE FOLLOWING PARAMETERS","≤≠",                              00029200
":MAX1MIN1MAX2MIN2MAXAMTNADPCONIKHVRA≤≠:←"),F10("≤≠:",6I4,4I1,          00029300
I2,I1,2I2,"←"),R6(6I4,4I1,I2,I1,2I2),CKPAR(                             00029400
"≤≠RECHECK PARAMETER VALUES. IF OK TYPE 0 ","ELSE TYPE 1.≤≠:←")00029500
,F11("≤≠≠TO GET STARTED,FIRST SET PROGRAM MODE.≤≠:←"),F12(              00029600
"≤≠ENTER EITHER A STANDARD TEST NUMBER ","OR ZERO.≤≠:←");SWITCH00029700
 FORMAT NAME1←("≤≠HI ",A6,A3,"←"),("≤≠HEY THERE ",A6,A3,"←"),(          00029800
"≤≠GOOD TO SEE YOU ",A6,A3,"←"),("≤≠HOWDY ",A6,A3,"←"),(                00029900
"≤≠HOW DO ",A6,A3,"←"),("≤≠WELCOME ",A6,A3,"←"),(                       00030000
"≤≠NICE TO SEE YOU ",A6,A3,"←"),("≤≠HELLO ",A6,A3,"←");LIST Z1          00030100
(MODE);LIST Z2 (FOR I←1 STEP 1 UNTIL 5 DO STD[1,I]);LIST Z5 (           00030200
FOR I←1 STEP 1 UNTIL 26 DO PAR[I]);LIST Z6 (K);LIST Z7 (FOR I←500030300
 STEP 2 UNTIL 9 DO[PAR[I],PAR[I+1]],FOR J←0 STEP 1 UNTIL 3 DO P00030400
[J],PAR[13],K,PAR[14],PAR[26]);LIST PK (PCK);                           00030500
PROCEDURE BADOP(P,FL);                                                  00030600
REAL ARRAY P[0];                                                        00030700
REAL FL;                                                                00030800
 BEGIN                                                                  00030900
    FORMAT BADOP1("≤≠ALL OP-CODES ARE ZERO.≤≠←"),BADOP2(                 00031000
    "≤≠OPERATION CODE VALUE EXCEEDS MAX OF 4.≤≠←");FL ←0;                00031100
    IF P[0]=0 AND P[1]=0 AND P[2]=0 AND P[3]=0 THEN                     00031200
    BEGIN                                                               00031300
COMMENT--CHECK FOR NO OP SELECTION;WOUT2(BADOP1);FL ←1;                  00031400
    END                                                                 00031500
    ELSE                                                                00031600
    IF P[0]>4 OR P[1]>4 OR P[2]>4 OR P[3]>4 THEN                        00031700
    BEGIN                                                               00031800
COMMENT--UNDEFINED OP SELECTION CHECK;WOUT2(BADOP2);FL ←1;               00031900
    END;                                                                00032000
 END BADOP;                                                             00032100
PROCEDURE BADLIM(U,FL);                                                 00032200
REAL ARRAY U[1];                                                        00032300
REAL FL;                                                                00032400
 BEGIN                                                                  00032500
    FORMAT BADLM("≤≠LOWER LIMIT",I2," IS GEQ TO UPPER LIMIT.≤≠←"00032600
    );                                                                  00032700
   REAL K;LIST LST (K);FL ←0;FOR K ←5 STEP 2 UNTIL 9 DO                 00032800
    IF U[K+1]≥U[K]THEN                                                  00032900
```

```
        BEGIN                                                                   00033000
           WOUT1(RADLIM,LST);FL ←1;                                             00033100
        END;                                                                    00033200
      END RADLI4;REM ←0;                                                        00033300
      IF SAVSTRT≠0 AND (MODE=2 OR MODE=4)AND MOD2FL≠0 THEN GO TO MOD2           00033400
      ;                                                                         00033500
COMMENT--SET TEST MODE;WOUT2(F11);RIN1(R1,Z1);                                  00033600
      IF MODE =3 AND SAVSTRT =0 THEN GO TO TESTPT;                              00033700
      IF (MODE=2 OR MODE=4)AND MOD2FL =0 THEN GO TO TESTID;                     00033800
      IF MODE=0 THEN GO TO TESTID;MOD2:PPL ←PPL+1;WOUT2(NAME);RIN1(             00033900
      NAME2,Z2);STRTVAL ←TIME(1)×PPL;N ←RANDOM (7,0,STRTVAL);WOUT3(             00034000
      NAME1,N,Z3);                                                              00034100
      IF (MODE =2 OR MODE =4)AND TID≥0 THEN WOUT1(F31,Z50);                     00034200
      IF MODE=1 OR MOD2FL=0 THEN WOUT2(F32);N←STRTVAL←0;                        00034300
      IF MOD2FL≠0 THEN GO TO TESTPT;TESTID:WOUT2(F12);                          00034400
COMMENT--SELECT STANDARD TEST OR CHOOSE CUSTOMIZED OPTION.;RIN1(R7,Z4);         00034500
      PAR[1]←2;TID ←OP[1]×10 +OP[2]-1;TUD ←TID+1;                               00034600
      IF TID≥0 THEN                                                             00034700
      BEGIN                                                                     00034800
        READ(PARAMETID],PAREOR,Z5)[GOON:BEG];                                   00034900
COMMENT--LOAD STANDARD TEST #TID PARAMETERS;GOON:FOR I←0 STEP 1 UNTIL 3          00035000
        DO P[I]←PAR[I+1];BADOP(P,FLAG);                                         00035100
        IF FLAG =1 THEN                                                         00035200
        BEGIN                                                                   00035300
          FLAG ←0;GO TO TESTID;                                                 00035400
        END;READ(PARAME[99+TID],RTIT,Z50)[GOGO:BEG];GOGO:                       00035500
        IF PAR[20]=2 AND MODE ≠0 THEN                                           00035600
        BEGIN                                                                   00035700
          LIMGON←(100-PAR[25])/PAR[24];LIMAST←PAR[25]/(100-PAR[24])             00035800
          ;                                                                     00035900
        END;BADLIM(PAR,FLAG);                                                   00036000
        IF FLAG =1 THEN                                                         00036100
        BEGIN                                                                   00036200
          FLAG ←0;GO TO TESTID                                                  00036300
        END;                                                                    00036400
OMMENT--SET ECHOCHECK ON ANSWERS FOR INTERACTIVE/STANDARD MODE;                 00036500
        IF MODE=2 OR MODE=4 THEN                                                00036600
        BEGIN                                                                   00036700
          K ←0;MOD2FL ←1;                                                       00036800
        END;                                                                    00036900
        IF MODE =1 OR MODE =4 THEN                                              00037000
```

```
      BEGIN                                                                    00037100
         WOUT2(F14);RTN1(R1,Z6);                                               00037200
      END;                                                                     00037300
      IF MODE =4 AND SAVSTRT =0 THEN                                           00037400
      BEGIN                                                                    00037500
         SAVSTRT←1;GO TO BEG                                                   00037600
      END;                                                                     00037700
      IF MODE=4 THEN GO TO BEG;                                                00037800
      IF MODE=2 AND PPL=0 AND SAVSTRT≠0 THEN GO TO BEG;GO TO                   00037900
      TESTPT:                                                                  00038000
   END STANDARD TEST SELECTION;                                                00038100
COMMENT--ENTER PARAMETERS FOR CUSTOMIZED TEST;OVER:WOUT2(F9);FOR I ←1           00038200
   STEP 1 UNTIL 26 DO PAR[I]←0;RIN1(R6,Z7);WOUT1(F10,77);PAR[11]←-             00038300
   1;WOUT2(C<PAR);RIN1(R1,PK);                                                 00038400
   IF PCK =1 THEN GO TO OVER;BADOP(P,FLAG);                                    00038500
   IF FLAG =1 THEN                                                             00038600
   BEGIN                                                                       00038700
      FLAG ←0;GO TO TESTID;                                                    00038800
   END;BADLI(PAR,FLAG);                                                        00038900
   IF FLAG =1 THEN                                                             00039000
   BEGIN                                                                       00039100
      FLAG ←0;GO TO TESTID                                                     00039200
   END;                                                                        00039300
   IF MODE=2 OR MODE=4 THEN                                                    00039400
   BEGIN                                                                       00039500
      IF MODE =4 AND SAVSTRT=0 THEN                                            00039600
      BEGIN                                                                    00039700
         SAVSTRT←1;GO TO SET2FL;                                               00039800
      END;IF MODE=2 THEN K←0;SET2FL:MOD2FL←1;                                  00039900
      IF SAVSTRT≠0 THEN GO TO BEG;                                             00040000
   END;                                                                        00040100
 END BLOCK 01:TESTPT:                                                          00040200
 BEGIN                                                                         00040300
COMMENT--BLOCK 03;LIST SV (SAVSTRT);LIST ZBAS (PART26);                        00040400
   LABEL OLDTEST,STRT,MOD3;FORMAT OV(I8),BASFOR(                               00040500
   "≤≠≠ALL PROBLEMS ARE BASE",I3,":←"),IV(                                     00040600
   "≤≠ENTER STARTING VALUE FOR OLD TEST≤≠:←");                                  00040700
   IF MODE =3 AND SAVSTRT =0 THEN GO TO MOD3:                                  00040800
   IF MODE =2 AND SAVSTRT =0 THEN GO TO STRT:                                  00040900
   IF MODE =0 THEN WOUT2(F1);WOUT2(F1);                                        00041000
COMMENT--INITIALIZE COUNTERS;                                                  00041100
```

```
IF MODE ≠0 THEN                                                    00041200
BEGIN                                                              00041300
   FOR I←1 STEP 1 UNTIL 25 DO                                      00041400
   BEGIN                                                           00041500
      FOR IT ←1 STEP 1 UNTIL 2 DO KEY[I,IT]←0;FOR J←1 STEP 1       00041600
      UNTIL 4 DO RPTR[I,J]←0;                                      00041700
   END;                                                            00041800
END;NUM←0;NRNG←0;FLAG←0;OPFLAG←0;RFLAG ←0;                          00041900
IF MODE =2 OR MOD3FL =1 THEN                                       00042000
BEGIN                                                              00042100
   STRTVAL ←SAVSTRT;GO TO OLDTEST;                                 00042200
END;MOD3;                                                          00042300
IF MODE =3 AND SAVSTRT =0 THEN                                     00042400
BEGIN                                                              00042500
   MOD3FL ←1;WOUT2(IV);RIN1(OV,SV);GO TO BEG;                      00042600
END OLDTEST SWITCH;START:STRTVAL ←TIME(1)×(TIME(4)+1);             00042700
IF STRTVAL <10*7 THEN STRTVAL ←STRTVAL +10*7;                      00042800
IF STRTVAL >10*8 THEN STRTVAL ←STRTVAL-(STRTVAL DIV (10*8))×10*    00042900
8;                                                                 00043000
IF MODE =2 AND SAVSTRT =0 THEN                                     00043100
BEGIN                                                              00043200
   SAVSTRT ←STRTVAL;GO TO BEG                                      00043300
END;OLDTEST;                                                       00043400
COMMENT--RECORD STRTVAL AND START RAND NUM GENERATOR,SET FINAL POSITIONS00043500
 OF POINTERS FOR INTERACTIVE MODE;                                 00043600
   IF MODE ≠0 THEN STOT[1,61←STRTVAL;                              00043700
   IF MODE = 2 THEN PAR[20]←0;                                     00043800
   IF MODE =0 THEN                                                 00043900
   BEGIN                                                           00044000
      WOUT1(64,255);WOUT2(NAME5)                                   00044100
   END;WOUT1(BASEOR,ZBAS);SVSTRT ←STRTVAL;STRTVAL←RANDOM(0,0,       00044200
   STRTVAL);STRTVAL←0;VPTR←OUTPTR←HURPTR←0;                        00044300
END BLOCK 03;BL4:                                                  00044400
BEGIN                                                              00044500
COMMENT--BLOCK 04;                                                 00044600
   LABEL LOOP,RANDOP,01,02,03,04,ITEMPRNT,RESP,ANSREAD,SPRT,SCR,   00044700
    EXIT;SWITCH OPSELCT←01,02,03,04;                               00044800
   REAL ROP,I1,T1,T2,T;OWN                                         00044900
   REAL ARRAY LAT[1:50];                                           00045000
   REAL ARRAY RR[1:4],RC[1:4];SWITCH FORMAT WR ←(I1),(I2),(I3),(I4)00045100
    ,(I5),(I6);LIST ZIA (A);LIST ZA1 (VPTR);SWITCH FORMAT TH ←(    00045200
```

```
        IF ANS ≥-99 THEN RI ← 2                                                 00049500
        ELSE                                                                    00049600
        IF ANS ≥-999 THEN RI ← 3;                                               00049700
      END RI;                                                                   00049800
      IF PAR[13]>25 THEN PAR[13]←25;                                            00049900
      IF PAR[20]=2 AND MODE≠0 AND TID≠0 THEN GO TO SPRT;                        00050000
      IF PAR[13]≤0 THEN PAR[13]←1;FOR II ←1 STEP 1 UNTIL PAR[13]DO              00050100
COMMENT--BEGIN ITEM GENERATION;                                                 00050110
      BEGIN                                                                     00050200
        SPRT:NUM←NUM+1;IF NUM > 25 THEN GO TO SCR;                              00050300
        LOOP:FOR I←5 STEP 2 UNTIL                                               00050310
        7 DO OP[(I-3)/2]←RANDOM(PAR[I],PAR[I+1],STRTVAL);                       00050400
        IF FLAG ≠0 THEN GO TO RANDOP;                                           00050500
        IF P[1]=0 THEN GO TO OPSELCT[P[0]]                                      00050600
        ELSE RANSLCT(P,FLAG,A1,B);RANDOP:                                       00050700
        IF OPFLAG ≠0 THEN GO TO OPSELCT[P[ROP]];ROP ←RANDOM (A1,B,B)            00050800
        ;GO TO OPSELCT[P[ROP]];01:                                              00050900
        BEGIN                                                                   00051000
COMMENT--BLOCK 041;                                                             00051100
          INTEGER N;OWN                                                         00051200
          REAL ARRAY VERTAD[1:4,0:9];LIST Z8 (VPTR,FOR J ←1 STEP 1              00051300
           UNTIL VPTR DO [VERTAD[J,0],VERTAD[J,1]]);LIST Z9 (VPTR,              00051400
           FOR J ←1 STEP 1 UNTIL VPTR DO VERTAD[J,I]);LIST Z41R (               00051500
           VPTR,FOR I←1 STEP 1 UNTIL VPTR DO [RR[VPTR],RC[VPTR],                00051600
           VERTAD[I,PAR[12]]]);FORMAT NOGOT(                                    00051700
           "S≠SELECTED TEST IS NOT OPERATIVE. TRY ANOTHER.←")};OPN ←00051800
           ;                                                                    00051900
           IF PAR[12]>2 THEN                                                    00052000
           BEGIN                                                                00052100
COMMENT--NONINTERACTIVE MULTIPLE ADDEND SEGMENT;                                00052200
             IF MODE ≠0 THEN                                                    00052300
             BEGIN                                                              00052400
               WOUT2(NOGOT);TRYCOL ←1;GO TO EXIT;                               00052500
             END;ANS ←0;VPTR ←VPTR +1;VERTAD[VPTR,0]←NUM;FOR I ←1                00052600
             STEP 1 UNTIL PAR[12]DO                                             00052700
             BEGIN                                                              00052800
               VERTAD[VPTR,I]←RANDOM(PAR[5],PAR[6],STRTVAL);ANS←                00052900
               ANS+VERTAD[VPTR,I];                                              00053000
             END;KEY[NUM,1]←ANS;KEY[NUM,2]←"←      ";RR[VPTR]←7 -RI(          00053100
             ANS);RC[VPTR]←RI(ANS)+2;                                           00053200
             IF VPTR =4 OR II =PAR[13]THEN                                      00053300
```

```
COMMENT--MULTIPLE ADDEND PRINTOUT;WOUT3(TV,0,Z8);FOR I ←2 STEP 1 UNTIL (00053500
                PAR[12]-1)DO WOUT3(TV,1,Z9);WOUT3(TV,2,Z41R);WOUT3(00053600
                TV,5,Z41);FOR K ←1 STEP 1 UNTIL 4 DO FOR J←0 STEP 100053700
                 UNTIL PAR[12]DO VERTAD[K,J]←0;WOUT2(LF);VPTR ←0;  00053800
              END MULTIPLE ADDEND OUTPUT;GO TO RESP;                00053900
           END MULTIPLE ADDEND SEGMENT;                             00054000
           IF PAR[11]≥0 THEN CARYCON(OP[1],OP[2],PAR[18],OPN,PAR[11]00054100
           ,PAR[19],PAR[17],ANS,PAR[26]);ANS←OP[1]+OP[2];           00054200
           IF ANS>PAR[9]OR ANS<PAR[10]THEN                          00054300
           BEGIN                                                    00054400
              OPFLAG←1;GO TO LOOP;                                  00054500
           END;OPFLAG←0;GO TO ITEMPRNT;                             00054600
        END BLOCK 041;02:                                           00054700
        BEGIN                                                       00054800
COMMENT--BLOCK 042;                                                 00054900
          INTEGER GUMMY;OPN ←1;                                     00055000
           IF PAR[11]≥0 THEN CARYCON(OP[1],OP[2],PAR[18],OPN,PAR[11]00055100
           ,PAR[19],PAR[17],ANS,PAR[26]);ANS←OP[1]-OP[2];           00055200
           IF ANS>PAR[9]OR ANS<PAR[10]THEN                          00055300
           BEGIN                                                    00055400
              OPFLAG←1;GO TO LOOP;                                  00055500
           END;OPFLAG←0;GO TO ITEMPRNT;                             00055600
        END BLOCK 042;03:                                           00055700
        BEGIN                                                       00055800
COMMENT--BLOCK 043;                                                 00055900
          INTEGER GUMMY;OPN ←2;                                     00056000
           IF PAR[11]≥0 THEN CARYCON(OP[1],OP[2],PAR[18],OPN,PAR[11]00056100
           ,PAR[19],PAR[17],ANS,PAR[26]);ANS←OP[1]×OP[2];           00056200
           IF ANS>PAR[9]OR ANS<PAR[10]THEN                          00056300
           BEGIN                                                    00056400
              OPFLAG←1;GO TO LOOP;                                  00056500
           END;OPFLAG←0;GO TO ITEMPRNT;                             00056600
        END BLOCK 043;04:                                           00056700
        BEGIN                                                       00056800
COMMENT--BLOCK 044:                                                 00056900
          INTEGER GUMMY;                                            00057000
           IF OP[1]=0 THEN                                          00057100
           BEGIN                                                    00057200
              OPFLAG←1;GO TO LOOP;                                  00057300
           END;OPFLAG←0;OPN ←3;                                     00057400
```

```
            IF ITD ≥0 THEN REM ←PAR[17]                                          00057500
            ELSE                                                                 00057600
COMMENT--NONINTERACTIVE NONSTANDARD TESTS PRODUCE REMAINDERS RANDOMLY;           00057700
            BEGIN                                                                00057800
               REM ←-1;PAR[15]←PAR[16]←9;PAR[11]←-1;                             00057900
            END;                                                                 00058000
            IF MODE ≠0 OR PAR[14]=0 THEN                                         00058100
COMMENT--INTERACTIVE MODE ALWAYS HAS ZERO REMAINDER PROBLEMS;                    00058200
COMMENT--HORIZONTAL FORM OF DIVISION ALWAYS HAS ZERO REMAINDER;                  00058300
            BEGIN                                                                00058400
               REM ←0;PAR[16]←PAR[15]←9;PAR[11]←-1;                              00058500
            END;ANS←OP[1]×OP[2];TEMP ←OP[1];OP[1]←ANS;                           00058600
            IF OP[1]>PAR[9]OR OP[1]<PAR[10]THEN                                  00058700
            BEGIN                                                                00058800
               OPFLAG ←1;GO TO LOOP;                                             00058900
            END;OPFLAG←0;ANS ←OP[2];OP[2]←TEMP;ZEROCK(ANS,PAR[15],PAR00059000
            [16],OP[2],OP[1],REM,PAR[26]);                                       00059100
            IF PAR[11]≥0 THEN CARYCON(OP[1],OP[2],PAR[18],OPN,PAR[11]00059200
            ,PAR[19],REM,ANS,PAR[26]);ANS←OP[1]DIV OP[2];REM←OP[1]MOD00059300
             OP[2];                                                              00059400
            IF ANS >PAR[7]OR ANS <PAR[8]THEN                                     00059500
            BEGIN                                                                00059600
               OPFLAG ←1;GO TO LOOP;                                             00059700
            END;OPFLAG←0;                                                        00059800
         END BLOCK 044;ITEMPRNT;                                                 00059900
         BEGIN                                                                   00060000
COMMENT--BLOCK 045 OUTPUT SECTION;                                               00060100
            REAL                                                                 00060200
            PROCEDURE BAS(AB,AA);                                                00060300
            REAL AA,AB;                                                          00060400
             BEGIN                                                               00060500
               REAL I,Y;                                                         00060600
               LABEL L1,L2;Y←0;I←0;L1:Y←Y+(AA MOD AB)×10*I;AA←AA DIV             00060700
                AB;                                                              00060800
               IF AA=0 THEN GO TO L2;I←I+1;GO TO L1;L2:BAS←Y;                    00060900
             END BAS;DWN                                                         00061000
            REAL ARRAY HORIZ[1:61],VERT[1:12];                                   00061100
            INTEGER N;                                                           00061200
            LABEL H1,H2;SWITCH FORMAT DIVV ←("≤≠",*(I2,"),",X13),"←"),00061300
            ("≤≠",*(X8,"--------",X2),"←"),("≤≠",*(X2,I5,X1,")",I*,X*)00061400
```

```
X*,"+",I*,"←"),("<≠",X2,X*,"-",I*,"←"),("≤≠",X2,X*,"X",I*00061600
,"←"),("≤≠",X5,"------<≠←"),("≤≠",I2,"),",X6,"------←"),(00061700
"≤≠",X4,I5,X1,")",I*,X*,"= ","←");LIST Z12 (NUM);LIST Z1300061800
 (HORIZ[1],HORIZ[2],RB[1],HORIZ[3]);LIST Z13R (RC[2],     00061900
HORIZ[4],HORIZ[5],RB[2],HORIZ[6]);LIST Z14 (OUTPTR,FOR I 00062000
←1 STEP 1 UNTIL 2×OUTPTR DO VERT[I]);LIST Z16 (OUTPTR,FOR00062100
 I ←1 STEP 2 UNTIL 2×OUTPTR-1 DO VERT[I]);LIST Z17 (      00062200
OUTPTR,FOR I ←1 STEP 1 UNTIL OUTPTR DO [VERT[I+8],RB[    00062300
OUTPTR],VERT[2×I],RC[OUTPTR]]);LIST Z10 (NUM,OP[1]);LIST 00062400
Z11 (OP[2]);LIST Z4R (OP[2],RB[1],OP[1],RC[1]);LIST Z42 (00062500
OUTPTR,FOR I ←1 STEP 1 UNTIL OUTPTR DO [RB[I],RC[I],VERT[00062600
I+8]]);LIST Z60 (OP[1],RB[1],OP[2]);LIST Z11R (RB[1],RB[200062700
],OP[2]);LIST ZPTR (OUTPTR);                             00062800
IF PAR[26]≠10 AND RFLAG =0 THEN                          00062900
BEGIN                                                    00063000
   OP[1]←BAS(PAR[26],OP[1]);OP[2]←BAS(PAR[26],OP[2]);REM 00063100
   ←BAS(PAR[26],REM);ANS←BAS(PAR[26],ANS);               00063200
END;                                                     00063300
IF MODE ≠0 THEN                                          00063400
BEGIN                                                    00063500
COMMENT--INTERACTIVE OUTPUT SEGMENT;                     00063600
   IF PAR[14]<0 THEN                                     00063700
   IF RANDOM(1,0,0)=1 THEN GO TO H1                      00063800
   ELSE GO TO H2;                                        00063900
   IF PAR[14]=0 THEN                                     00064000
   BEGIN                                                 00064100
COMMENT--HORIZONTAL INTERACTIVE OUTPUT;H1:RB[1]←RI(OP[2])+1;HV ←0;  00064200
      IF NUM =1 THEN WOUT2(LF);WOUT3(OPF,4,712);WOUT3(OPF00064300
      ,OPN,Z60);GO TO ANSREAD;                           00064400
   END HORIZONTAL INTERACTIVE OUTPUT;                    00064500
   IF PAR[14]>0 THEN                                     00064600
   BEGIN                                                 00064700
COMMENT--VERTICAL INTERACTIVE OUTPUT;H2:HV ←1;            00064900
      IF OPN ≠3 THEN                                     00064900
      BEGIN                                              00065000
         RB[1]←6-RI(OP[1]);RB[2]←8 -RB[1];WOUT2(LF);WOUT300065100
         (V1,0,Z10);WOUT3(V1,OPN+1,711R);WOUT4(V1,4);GO  00065200
         TO ANSREAD;                                     00065300
      END                                                00065400
      ELSE                                               00065500
      BEGIN                                              00065600
```

```
COMMENT--DIVISION OUTPUT;RB[1]←RT(OP[1])+2;RC[1]←6 -RB[1];WOUT3(V1,5,71200065700
          );WOUT3(V1,6,74R);GO TO ANSREAD;                          00065800
        END DIVISION OUTPUT;                                        00065900
      END VERTICAL INTERACTIVE OUTPUT;                              00066000
    END                                                             00066100
    ELSE                                                            00066200
    BEGIN                                                           00066300
COMMENT--NONINTERACTIVE OUTPUT;                                     00066400
      IF PAR[14]=0 THEN                                             00066500
      BEGIN                                                         00066600
COMMENT--HORIZONTAL NONINTERACTIVE OUTPUT;KEY[NUM,1]←ANS;KEY[NUM,2]← 00066700
        "←      ";HORPTR←HORPTR+1;HORIZ[3×HORPTR-2]←NUM;            00066800
        HORIZ[3×HORPTR-1]←OP[1];HORIZ[3×HORPTR]←OP[2];RB[           00066900
        HORPTR]←1 +RI(OP[2]);                                       00067000
        IF HORPTR =1 THEN RC[2]←11-RB[1];                           00067100
        IF HORPTR =2 OR IT =PAR[13]THEN                             00067200
        BEGIN                                                       00067300
          FOR J←0 STEP 1 UNTIL HORPTR -1 DO                         00067400
          BEGIN                                                     00067500
            IF J =0 THEN WOUT3(TH,2×OPN+J,Z13)                      00067600
            ELSE WOUT3(TH,2×OPN+J,713R);                            00067700
          END HORIZONTAL NONINTERACTIVE WRITE;WOUT2(F1);            00067800
          FOR I ←1 STEP 1 UNTIL 6 DO HORIZ[I]←0;HORPTR ←0;          00067900
                                                                    00068000
        END;GO TO RESP;                                             00068100
      END                                                           00068200
      ELSE                                                          00068300
      BEGIN                                                         00068400
COMMENT--VERTICAL NONINTERACTIVE OUTPUT;KEY[NUM,1]←ANS;             00068500
        IF REM =0 THEN KEY[NUM,2]←"←      "                       00068600
        ELSE KEY[NUM,2]←REM;OUTPTR ←OUTPTR +1;VERT[2×OUTPTR         00068700
         -1]←NUM;VERT[2×OUTPTR]←OP[1];VERT[8+OUTPTR]←OP[2];         00068800
                                                                    00068900
        IF OPN ≠3 THEN                                              00069000
        BEGIN                                                       00069100
          RB[OUTPTR]←7-RT(OP[1]);RC[OUTPTR]←2+RI(OP[1])             00069200
        END                                                         00069300
        ELSE                                                        00069400
        BEGIN                                                       00069500
          RB[OUTPTR]←2+RI(OP[1]);RC[OUTPTR]←8-RB[OUTPTR]            00069600
        END;                                                        00069700
```

```
                IF (OUTPTR =4 OR TT =PAR[13])AND OPN ≠3 THEN          00069800
                BEGIN                                                 00069900
                   WOUT3(TV,0,Z14);WOUT3(TV,OPN+2,Z42);WOUT3(TV,5,    00070000
                   ZPTR);WOUT2(F1);WOUT2(F1);WOUT2(F1);FOR I ←1        00070100
                   STEP 1 UNTIL 12 DO VERT[I]←0;OUTPTR ←0;GO TO       00070200
                   RESP;                                              00070300
                END NONDIVISION VERTICAL NONINTERACTIVE OUTPUT;       00070400
                IF (OUTPTR =4 OR TT =PAR[13])AND OPN =3 THEN          00070500
                BEGIN                                                 00070600
COMMENT--NONINTERACTIVE DIVISION OUTPUT;WOUT3(DIVV,0,Z16);WOUT3(DIVV,1, 00070700
                   ZPTR);WOUT3(DIVV,2,Z17);WOUT2(F1);WOUT2(F1);FOR    00070800
                   I ←1 STEP 1 UNTIL 12 DO VERT[I]←0;OUTPTR ←0;GO     00070900
                   TO RESP;                                           00071000
                END DIVISION VERTICAL NONINTERACTIVE OUTPUT;          00071100
             END VERTICAL NONINTERACTIVE OUTPUT;GO TO RESP;           00071200
          END NONINTERACTIVE OUTPUT;GO TO RESP;                       00071300
       END BLOCK 045;ANSREAD:                                         00071400
       BEGIN                                                          00071500
COMMENT--BLOCK 046 ANSWER READ;FORMAT VANS(X*,"←"),FR(I*,"←");LIST Z61 (00071600
       RB[2]);LIST Z62 (RC[1]);LIST Z19 (ANS);LIST Z19R (RB[1],       00071700
       RI(ANS)+1,ANS);LIST Z19A (RI(ANS)+1,ANS);RB[1]←RI(ANS);RC      00071800
       [1]←10-RB[1];                                                  00071900
       IF HV=1 THEN                                                   00072000
       BEGIN                                                          00072100
          IF OPN =3 THEN RB[2]←19;                                    00072200
          IF OPN ≠3 THEN WOUT1(VANS,Z62);                             00072300
       END;WHEN(3);T1 ←TIME(1);RIN3(WR,RB[1],Z18);T2 ←TIME(1)-(       00072400
       RB[1]+2)×12;LAT[NUM]←(T2-T1)/60;                               00072500
       IF A ≠ANS AND RFLAG =1 THEN                                    00072600
       BEGIN                                                          00072700
          WOUT1(F18,Z19);GO TO ITEMPRNT;                              00072800
       END SECOND ERROR RESPONSE;                                     00072900
       IF A =ANS AND RFLAG =1 THEN GO TO EXIT;                        00073000
       IF X =0 THEN GO TO SCR;                                        00073100
       IF HV=1 THEN                                                   00073200
       BEGIN                                                          00073300
          IF OPN ≠3 THEN WOUT1(VANS,Z62)                              00073400
          ELSE WOUT1(VANS,Z61);WOUT1(FR,Z19A);WOUT2(LF);GO TO         00073500
          SCR;                                                        00073600
       END                                                            00073700
       ELSE RB[1]←19+RI(OP[2]);WOUT3(OPF,5,Z19R);WOUT2(LF);           00073800
```

```
        END BLOCK 046;SCR;                                                  00073900
        BEGIN                                                               00074000
COMMENT--ANSWER/RESPONSE COMPARE BLOCK 048;                                 00074100
          LABEL MAS,NM;FORMAT MAST("S≠WELL DONE ",A6,A3,                    00074200
           ",YOU HAVE SHOWN MASTERY IN THIS TEST.←")",NONMST(               00074300
           "S≠THANK YOU ",A6,A3,                                            00074400
           ",YOU SHOULD WORK FURTHER ON PROBLEMS","LIKE THESE:S≠←")          00074500
           ,MERR(                                                           00074600
           "S≠PROBLEMS ON WHICH ERRORS WERE MADE ARE REPEATED BELOW"        00074700
           ,":S≠←");                                                        00074800
           IF NUM>25 AND WRNG≥PAR[21] THEN GO TO NM;                        00074900
           IF NUM>25 AND WRNG<PAR[21] THEN GO TO MAS;                       00075000
           IF MODE =2 AND A =ANS THEN ITRESP[NUM]←1                         00075100
           ELSE                                                             00075200
           IF MODE =2 AND A ≠ANS THEN ITRESP[NUM]←0;                        00075300
           IF A≠ANS AND RFLAG =0 THEN                                       00075400
           BEGIN                                                            00075500
              WRNG ←WRNG +1;RPTR[WRNG,1]←OP[1];RPTR[WRNG,2]←OPN;RPTR        00075600
              [WRNG,3]←OP[2];RPTR[WRNG,4]←ANS;RPTR[WRNG,0]←NUM;RPTR[        00075700
              WRNG,5]←HV;                                                   00075800
           END;                                                             00075900
           IF PAR[20]=1 THEN                                                00076000
           IF WRNG =PAR[21]THEN                                             00076100
           BEGIN                                                            00076200
              WOUT1(NONMST,Z3);MFL←1;GO TO EXIT                             00076300
           END                                                              00076400
           ELSE                                                             00076500
           IF NUM-WRNG=PAR[13]-PAR[21]+1 THEN                               00076600
           BEGIN                                                            00076700
              WOUT1(MAST ,Z3);                                              00076800
              IF WRNG≠0 THEN WOUT2(MERR);MFL ←0;GO TO EXIT                  00076900
           END;                                                             00077000
           IF PAR[20]=2 THEN                                                00077100
           BEGIN                                                            00077200
              L ←(((PAR[23]/100)*WRNG)×((1-                                 00077300
              PAR[23]/100)*(NUM-WRNG)))/(((PAR[22]/100)*WRNG)×((1-          00077400
              PAR[22]/100)*(NUM-WRNG)));                                    00077500
              IF L >LIMNON THEN                                             00077600
              BEGIN                                                         00077700
                 NM:WOUT1(NONMST,Z3);MFL←1;GO TO EXIT;                      00077800
```

```
                IF L <LIMAST THEN                                                00078000
                BEGIN                                                            00078100
                   MAS:WOUT1(MAST ,Z3);                                          00078200
                   IF WRNG≠0 THEN WOUT2(MERR);NFL←0;GO TO EXIT                   00078300
                END;GO TO BL4;                                                   00078400
             END;                                                                00078500
          END BLOCK 048;RESP:                                                    00078600
       END OF ITEM GENERATION LOOP;EXIT:                                         00078700
       BEGIN                                                                     00078800
COMMENT--BLOCK 049 WRAPUP AND EXIT;                                              00078900
*        FILE SCORES DISK SERIAL "GIESE" "CMS"(1,10,150,SAVE 30);                00079000
                                                                                 00079100
*       OWN REAL ARRAY ZETA[1:99];OWN                                            00079200
        INTEGER WRNGI,EF;                                                        00079300
        INTEGER N,NUMB;FORMAT TIM(                                               00079400
         "≤≠TIME IS NEARLY UP,QUIT NORMALLY AND REEXECUTE.≤≠←"),FR16(             00079500
         "≤≠",X2,I3,X2,I7,"←"),F16("≤≠",X2,I3,X2,I7,X1,A6,"←"),F70(              00079600
         "≤≠","IF FINISHED,ENTER 0 ELSE ENTER 1≤≠:←"),F15(                       00079700
         "≤≠ANSWER KEY:  TEST #",I3,I8,"≤≠≠",                                    00079800
         "PROBLEM ANSWER REMAINDER≤≠←",→,TEAROF("≤≠OK ",A6,A3,                   00079900
         ",PLEASE TEAR OFF PAPER AT"," DOTTED LINE.←"),DOTLIN(                   00080000
         "≤≠-----------------------------------------------←"),                  00080100
         DATAREC(A6,2(A6,A3),I8,3(X2,I2,X2),X2,I2,X1,A6,X2,I3,X1,F4.1,00080200
         ),ITMOUT(A6,2(A6,A3),*I1),PPLOUT("≤≠",A6,A3,2A6,A3,"←"),HDR(00080300
         "≤≠≠L  A  S  T  ---FIRST ---STARTVAL ITEMS WRONG",                      00080400
         " RIGHT TST# DATE  PREC LINC ←"),PP8("≤≠",                              00080500
         "NUMBER OF ERRORS =",I2,"≤≠","AVERAGE RESPONSE TIME = ",F6.2,00080600
         ," SEC.","≤≠","PERCENTAGE SCORE =",F5.1,"←");                           00080700
       LABEL AGAIN,NEXT,AG,TIMOUT;LIST Z20 (WRNG,LATBAR[1],N);LIST               00080800
        Z22 (FOR I←3 STEP 1 UNTIL 5 DO STD[J,I],STD[J,1],STD[J,2],              00080900
        FOR I←6 STEP 1 UNTIL 11 DO STD[J,I],ZETA[J],LATBAR[J]);LIST             00081000
        Z23 (I,KEY[I,1],KEY[I,2]);LIST Z40 (I,KEY[I,1]);LIST Z51 (              00081100
        STD[1,3],STD[1,4],STD[1,5],STD[1,1],STD[1,2],NUMB,                      00081200
        FOR I←1 STEP 1 UNTIL NUMB DO ITRESP[I]);LIST Z52 (FOR I←1               00081300
        STEP 1 UNTIL 5 DO STD[1,I]);LIST Z24 (PTR);LIST Z56 (TUD,               00081400
        SVSTRT);                                                                00081500
        IF RFLAG=0 THEN NUMB ←NUM;                                              00081600
        IF TRYCOL=1 THEN                                                        00081700
        BEGIN                                                                   00081800
           TRYCOL←0;GO TO AG;                                                   00081900
        END;                                                                    00082000
```

```
IF MODE ≠0 THEN                                                    00082100
BEGIN                                                              00082200
   IF WRNG ≠0 THEN                                                 00082300
   BEGIN                                                           00082400
      IF RFLAG =1 THEN GO TO NEXT;WRNGI ←1;RFLAG ←1;AGAIN:OP00082500
      I]←RPTR[WRNGI,1];OPN ←RPTR[WRNGI,2];OP[2]←RPTR[WRNGI,00082600
      3];ANS ←RPTR[WRNGI,4];PAR[14]←RPTR[WRNGI,5];NUM ←RPTR[00082700
      RNGI,0];GO TO ITEMPRNT;NEXT:WRNGI ←WRNGI +1;                 00082800
      IF WRNGI <WRNG THEN GO TO AGAIN;                             00082900
   END                                                             00083000
   IF ERROR SEGMENT;                                               00083100
   IF MODE =2 AND PAR[20]=0 THEN                                   00083200
   BEGIN                                                           00083300
      WRITE(ITEMSCR[PT],ITMOUT,Z51)[EOF];PT ←PT+1;CLOSE(           00083400
      ITEMSCR,SAVE);                                               00083500
   END;LATBAR[1]←0;FOR I ←1 STEP 1 UNTIL NUMB DO LATBAR[           00083600
   1]←LATBAR[1]+LAT[I];LATBAR[1]←LATBAR[1]/NUMB;ZETA               00083700
   [1]←N←((NUMB-WRNG)/NUMB)×100;STD[1,7]←NUMB;STD[1,8]             00083800
   ←WRNG;STD[1,9]←NUMB -WRNG;STD[1,10]←TUD;STD[1,11]←              00083900
   TIME(0);                                                        00084000
   IF TID ≥0 THEN WOUT1(TITLE,Z50);                                00084100
   IF TID ≥0 THEN WOUT1(F13,Z56);WOUT1(PPLOUT,Z52);WOUT1(PP800084200
   ,Z2));WOUT1(TEAROF,Z3);J ←1;WOUT2(DOTLIN);FOR I←1 STEP          00084300
   1 UNTIL 5 DO WOUT2(F1);READ (SCORES[01,R4,PTR);WRITE(           00084400
   SCORES[PTR],DATAREC,Z22)[EOF];PTR ←PTR+1;WRITE(SCORES[0],00084500
   R4,PTR);CLOSE(SCORES,SAVE);                                     00084600
   IF PAR[20]≠0 THEN                                               00084700
   IF NFL=1 THEN                                                   00084800
   BEGIN                                                           00084900
      READ(NONMAST[01,R4,NPT)[EOF];WRITE(NONMAST[NPT],             00085000
      DATAREC,Z22)[EOF];NPT←NPT+1;WRITE(NONMAST[0],R4,NPT)[        00085100
      EOF];CLOSE(NONMAST,SAVE);                                    00085200
   END                                                             00085300
   ELSE                                                            00085400
   BEGIN                                                           00085500
      READ(MASTER[0],R4,MPT)[EOF];WRITE(MASTER[MPT],DATAREC,00085600
      Z22)[EOF];MPT←MPT+1;WRITE(MASTER[0],R4,MPT)[EOF];CLOSE00085700
      (MASTER,SAVE);                                               00085800
   END;                                                            00085900
   IF TIME(2)≥6800 THEN                                            00086000
   BEGIN                                                           00086100
```

```
            TP←1;GO TO TIMOUT;                                              00086200
          END;RIN1(R1,Z21);                                                 00086300
COMMENT--READ IN SWITCH VALUE TO RUN AGAIN OR STOP;AG;WOUT2(F70);RIN1(R100086400
          ,Z21);                                                            00086500
          IF RPT ≠0 THEN GO TO BEG;TIMOUT:                                  00086600
          IF TP=1 THEN WOUT2(TIM);                                          00086700
          IF MODE =2 THEN                                                   00086800
          BEGIN                                                             00086900
             WRITE(ITEMSCR[2],R2,PT-3,NUMB,SAVSTRT,TIME(0));WRITE(          00087000
             ITEMSCR[1],RTIT,Z50);WRITE(ITEMSCR[0],R4,PT);CLOSE(            00087100
             ITEMSCR,SAVE);                                                 00087200
          END;                                                              00087300
        END                                                                 00087400
        ELSE                                                                00087500
        BEGIN                                                               00087600
COMMENT--START MODE ZERO OUTPUT;                                            00087700
COMMENT--START PRINTOUT OF TEST KEY,MODE ZERO;FOR I ←1 STEP 1 UNTIL 5 D0000087800
          WOUT2(F1);                                                        00087900
          IF TID ≥0 THEN WOUT1(TITLE,Z50);WOUT1(F15,Z56);FOR I←1            00088000
          STEP 1 UNTIL PAR[13]DO                                            00088100
          IF KEY[I,2]="←      "THEN WOUT1(F16,Z23)                          00088200
          ELSE WOUT1(FR16,Z23);WOUT2(F70);RIN1(R1,Z21);                     00088300
          IF RPT ≠0 THEN GO TO BEG;                                         00088400
        END ZERO MODE;                                                      00088500
      END BLOCK 049WRAPUP AND EXIT;                                         00088600
    END BLOCK 04 ;EOF;WOUT2(F71);RIN1(R1,Z21);                              00088700
    IF RPT ≠0 THEN GO TO STRTR;                                             00088800
  END BLOCK 00 MAIN;                                                        00088900
END.                                                                        00089000
```

-WAIT...

SAMPLE CONTENTS OF STAN/CMS: STANDARD-TEST PARAMETER FILE

PARAMETER FORMAT IS AS FOLLOWS (CF. CH.4 FOR DESCRIPTION):

```
OPCDUPL1-LOL1-UPL2-LOL2-UPL3----LOL3----CBONIHVABRMDMSCRZ1Z2ALBTBA
1        09     0     9     0       18       0-1210 099 01120202101010l0
2        18     0     9     0        9       0-1210 099 0112020210101010
3        09     0     9     0       81       0-1210 099 0112020210101010
4        09     1     9     0       81       0-1210 099 0112020210101010
1       899   101   899   101      999     202 0205 199 0312021050100510
1        99    11    99    11      198     110 1205 199 0212021050100510
1      9999  1001  9999  1001    19998    2002-1205 199 0112021050100510
1        99    11    99    11      396      44-1405 199 0112021050100510
2       999   201   899   101      999     101 0205 199 0312021050100510
2        99    21    89    11       88      10 1205 199 0212021050100510
2      9999  2001  8999  1001     8999    1001-1205 199 0112021050100510
3       999   101     9     2      999     202 0205 199 0312021050100510
3        99    11    99    22     9801    1210 1205 199 0222021050100510
3      9999  1001   999   101  9989001   10111-1205 199 0112021050100510
4         9     2   888   111      999     222 0205 199 0312001050100510
4         9     2  4999   111     9999    1001 1205 199-1412001050100510
4        09     2   999   101      999     101-1205 122 0112021050100510
4        99    11    99    10      999     101-1205 199 0112021050100510
4       999   101    99     1     9999    1001-1205 199 1112021050100510
1234     99     5    99     5       99       1-1210-1990011000000000010
```

TESTS CORRESPONDING TO THE ABOVE PARAMETERS ARE LABELED AS FOLLOWS:

BASIC ADDITION FACTS (0-9): TEST #1.
BASIC SUBTRACTION FACTS (0-9): TEST #2.
BASIC MULTIPLICATION FACTS (0-9): TEST #3.
BASIC DIVISION FACTS (0-9): TEST #4.
3-DIGIT ADDITION WITHOUT REGROUPING: TEST #5.
2-DIGIT ADDITION WITH REGROUPING: TEST #6.
4-DIGIT ADDITION: TEST #7.
2-DIGIT COLUMN ADDITION, 4 ADDENDS: TEST #8.
3-DIGIT SUBTRACTION WITHOUT REGROUPING: TEST #9.
2-DIGIT SUBTRACTION WITH REGROUPING: TEST #10.
4-DIGIT SUBTRACTION: TEST #11.
3 BY 1 DIGIT MULTIPLICATION WITHOUT REGROUPING: TEST #12.
2 BY 2 DIGIT MULTIPLICATION WITH REGROUPING: TEST #13.
4 BY 3 DIGIT MULTIPLICATION: TEST #14.
3 BY 1 DIGIT DIVISION WITHOUT REGROUPING: TEST #15.
4 BY 1 DIGIT DIVISION WITH REGROUPING: TEST #16.
3 BY 1 DIGIT DIVISION,ZERO IN QUOTIENT: TEST #17.
3 BY 2 DIGIT DIVISION,NO REMAINDERS: TEST #18.
DIVISION BY 3-DIGIT DIVISORS,REMAINDERS NOT ZERO: TEST #19.
MISCELLANEOUS PRACTICE PROBLEMS: PRACTICE PAGE #20.

80:

```
 1←

     NAME = FIRST----L--A--S--T-----
            TESTING  ONE                ←

HOWDY TESTING
SELECT A TEST AS FOLLOWS:
ENTER EITHER A STANDARD TEST NUMBER OR ZERO.
:20←

ENTER 1 IF CORRECT ANSWERS ARE TO BE PRINTED ELSE ENTER ZERO.
:←



ALL PROBLEMS ARE BASE 10:
 1).       ------
        12 ) 72    = 6←

 2).        7  X  6 = 42←


 3).       66
         - 23
       ------
           43←

 4).      90  /  5 = 18←

 5).       9  X  9 = 81←


 6).       48
         + 33
       ------
           81←


 7).       63
         - 33
       ------
           30←


 8).       31
         + 11
       ------
           42←

 9).        ------
         8 ) 64    = 8←

10).      11  X  6 = 66←

MISCELLANEOUS PRACTICE PROBLEMS: PRACTICE PAGE #20.
TEST # 2020981850
TESTING  ONE
NUMBER OF ERRORS = 0
AVERAGE RESPONSE TIME =    2.59 SEC.
PERCENTAGE SCORE =100.0
OK TESTING  ,PLEASE TEAR OFF PAPER AT DOTTED LINE.
-----------------------------------------------
```

```
  1←

     NAME = FIRST----L--A--S--T-----
            TESTING  LAST              ←

HELLO TESTING
SELECT A TEST AS FOLLOWS:
ENTER EITHER A STANDARD TEST NUMBER OR ZERO.
:50←

ENTER 1 IF CORRECT ANSWERS ARE TO BE PRINTED ELSE ENTER ZERO.
:0←



ALL PROBLEMS ARE BASE 10:

 1).     7  +  1 = 8←

 2).     1  +  1 = 1←

 3).     0  +  9 = 9←

 4).     5  +  1 = 6←

 5).     2  +  5 = 7←

 6).     7  +  0 = 7←

 7).     0  +  9 = 9←

 8).     9  +  8 = 17←

 9).     0  +  9 = 9←

WELL DONE TESTING  .YOU HAVE SHOWN MASTERY IN THIS TEST.
PROBLEMS ON WHICH ERRORS WERE MADE ARE REPEATED BELOW:

 2).     1  +  1 = 2←

BASIC ADDITION FACTS (0-9):  TEST #1.
TEST # 5014220556
TESTING  LAST
NUMBER OF ERRORS = 1
AVERAGE RESPONSE TIME =    0.88 SEC.
PERCENTAGE SCORE = 89.0
OK TESTING  ,PLEASE TEAR OFF PAPER AT DOTTED LINE.
-------------------------------------------------
```

```
←

     NAME = FIRST----L--A--S--T-----
            TESTING  TWO                ←

HOWDY TESTING
LETS GO WITH
3 BY 1 DIGIT MULTIPLICATION WITHOUT REGROUPING: TEST #12.


ALL PROBLEMS ARE BASE 10:

 1).     233
       X   3
      ------
         698←


 2).     212
       X   4
      ------
         456←

THANK YOU TESTING  .YOU SHOULD WORK FURTHER ON PROBLEMS LIKE THESE:


 1).     233
       X   3
      ------
         699←


 2).     212
       X   4
      ------
         844←

THE CORRECT ANSWER IS  848
TYPE IN THE CORRECT ANSWER

 2).     212
       X   4
      ------
         848←

3 BY 1 DIGIT MULTIPLICATION WITHOUT REGROUPING: TEST #12.
TEST # 1227076664
TESTING  TWO
NUMBER OF ERRORS = 2
AVERAGE RESPONSE TIME =    3.16 SEC.
PERCENTAGE SCORE =  0.0
OK TESTING  ,PLEASE TEAR OFF PAPER AT DOTTED LINE.
-------------------------------------------------
```

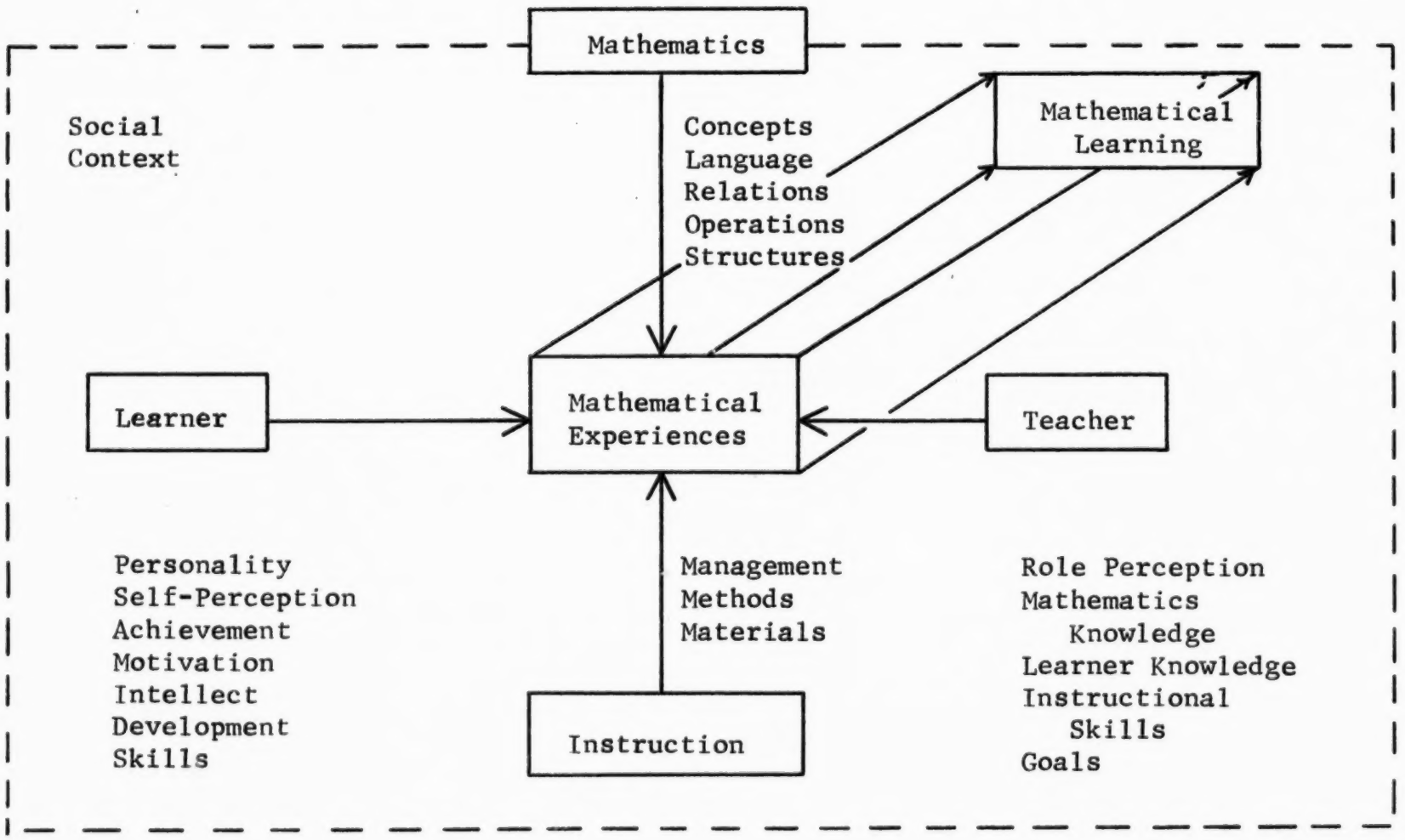
*LIST GIESE/CMS NO-WAIT...
SAMPLE CONTENT OF GIESE/CMS: STUDENT HISTORY FILE

NOTE: RECORD #1 CONTAINS A 4-DIGIT NUMBER WHICH POINTS TO
THE NEXT AVAILABLE RECORD FOR PUPIL DATA ENTRY.

```
  12
L---A---S---T--FIRST----STARTVAL ITEMS WRONG RIGHT TST# DATE  PRFC LTNC
KNOP            DONNA    92483820    5     0     5    30 069107  100 34.5
NIELSEN         BONNIE   92483820    5     0     5    30 069106  100 10.4
DERSIEN         AMY      92483820    5     0     5    30 069107  100  9.1
HANSEN          SKIP     92483820    5     1     4    30 069107   80  7.3
POLZIN          LARRY    71918126   47     5    42     1 069107   89  2.7
GREENWOOD       JOHN     81410824   16     3    13     1 069107   81  6.1
DEGRAVE         PAT      19770502   16     0    16     1 069107  100  2.6
CASPER          TIM      24225534   31     1    30     1 069107   97  3.5
COCA            SAM      10481024   31     1    30     1 069107   97  5.6
      80:
```

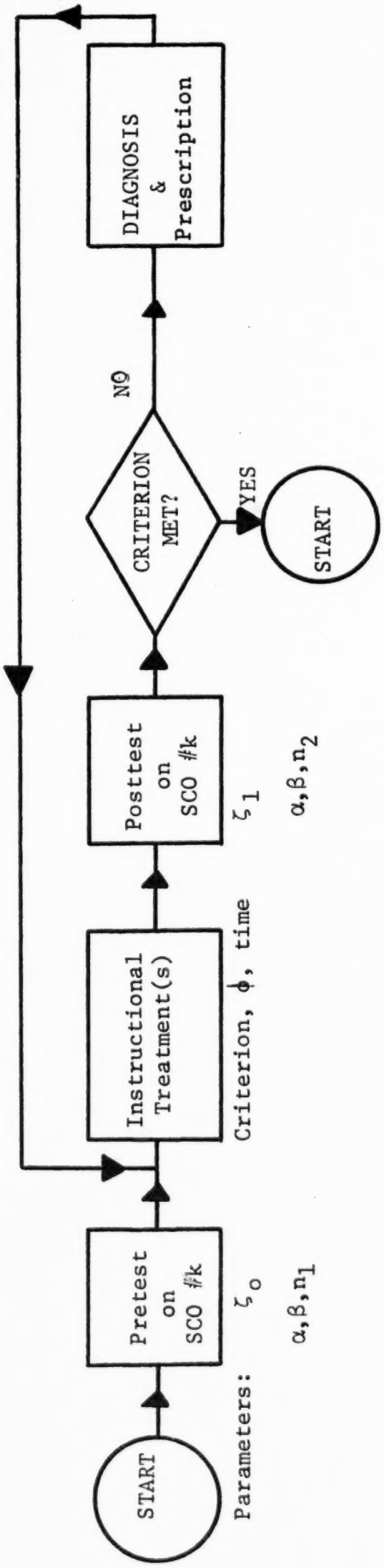
```
        *LIST GIESE/CMS NO←
  11
AGAIN           TESTING   27076664   5     0     5     12 069134  100   2.9
MEMORY          CORA      16167392   25    10    15     1 069134   60  -0.2
KIN             MANNY     98953204   2     2     0      1 069134    0   2.7
LINE            BEA       84929064   3     2     1      1 069134   33   3.6
CUSTOM          DESIGN     2219128   1     0     1      0 069134  100   3.2
LAST            TESTING   14220556   9     1     8     50 069134   89   0.9
FULLY           HOPE      28516920   2     2     0      2 069134    0   3.6
BROOK           DONNY     14240980   26    1     25     3 069134   96   0.2
GOOD            AMI       68399026   3     2     1      4 069134   33   4.2
RANGER          LONE      29350124   16    0     16     1 069134  100   0.0
      80:*LIST GIESEN/CMS NO←
   5
KIN             MANNY     98953204   2     2     0      1 069134    0   2.7
LINE            BEA       84929064   3     2     1      1 069134   33   3.6
FULLY           HOPE      28516920   2     2     0      2 069134    0   3.6
GOOD            AMI       68399026   3     2     1      4 069134   33   4.2
      80:*LIST GIESEM/CMS NO←
   5
AGAIN           TESTING   27076664   5     0     5     12 069134  100   2.9
LAST            TESTING   14220556   9     1     8     50 069134   89   0.9
BROOK           DONNY     14240980   26    1     25     3 069134   96   0.2
RANGER          LONE      29350124   16    0     16     1 069134  100   0.0
        *LIST ITEMS/CMS NO←
   8
BASIC ADDITION FACTS (0-9):  TEST #1.
 525161673920 69134
MEMORY          CORA      0100000000111111111101111
9000            HAL       1101111010011111101111011
5500            BURROUGHS 1111111111111111111111110
360             IBM       1111111111101111111111111
3600            CDC       0111111111111111111111111
      80:
```

APPENDIX G



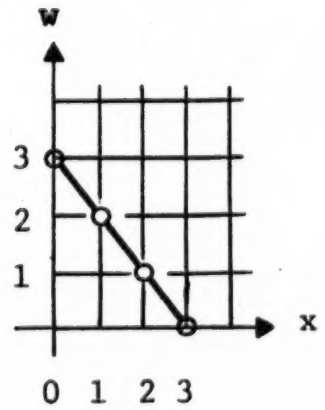
A GENERAL MODEL FOR THE ESM CURRICULUM

Figure 1



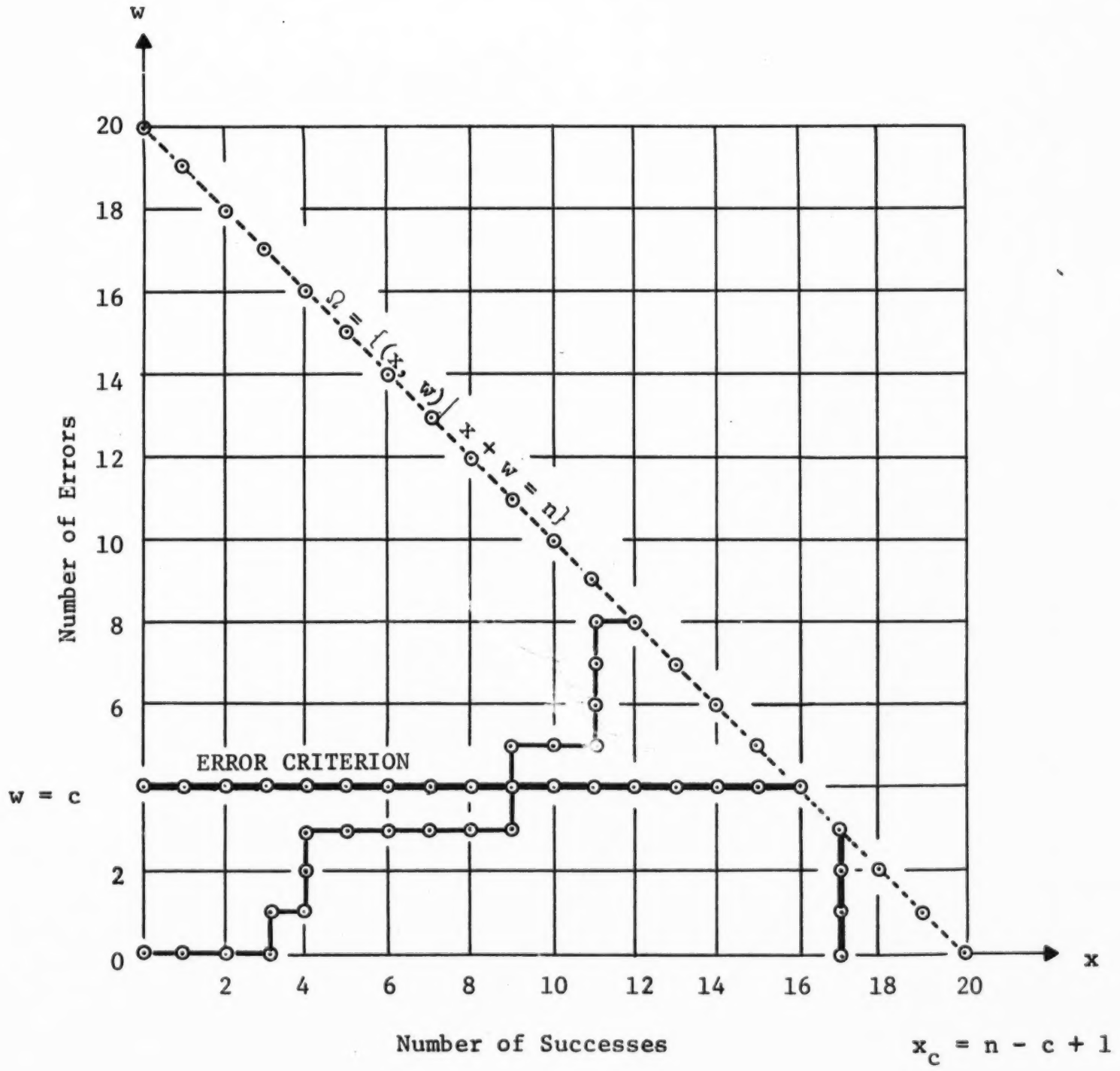
A Pattern for Instruction

Figure 2



Lattice Diagram

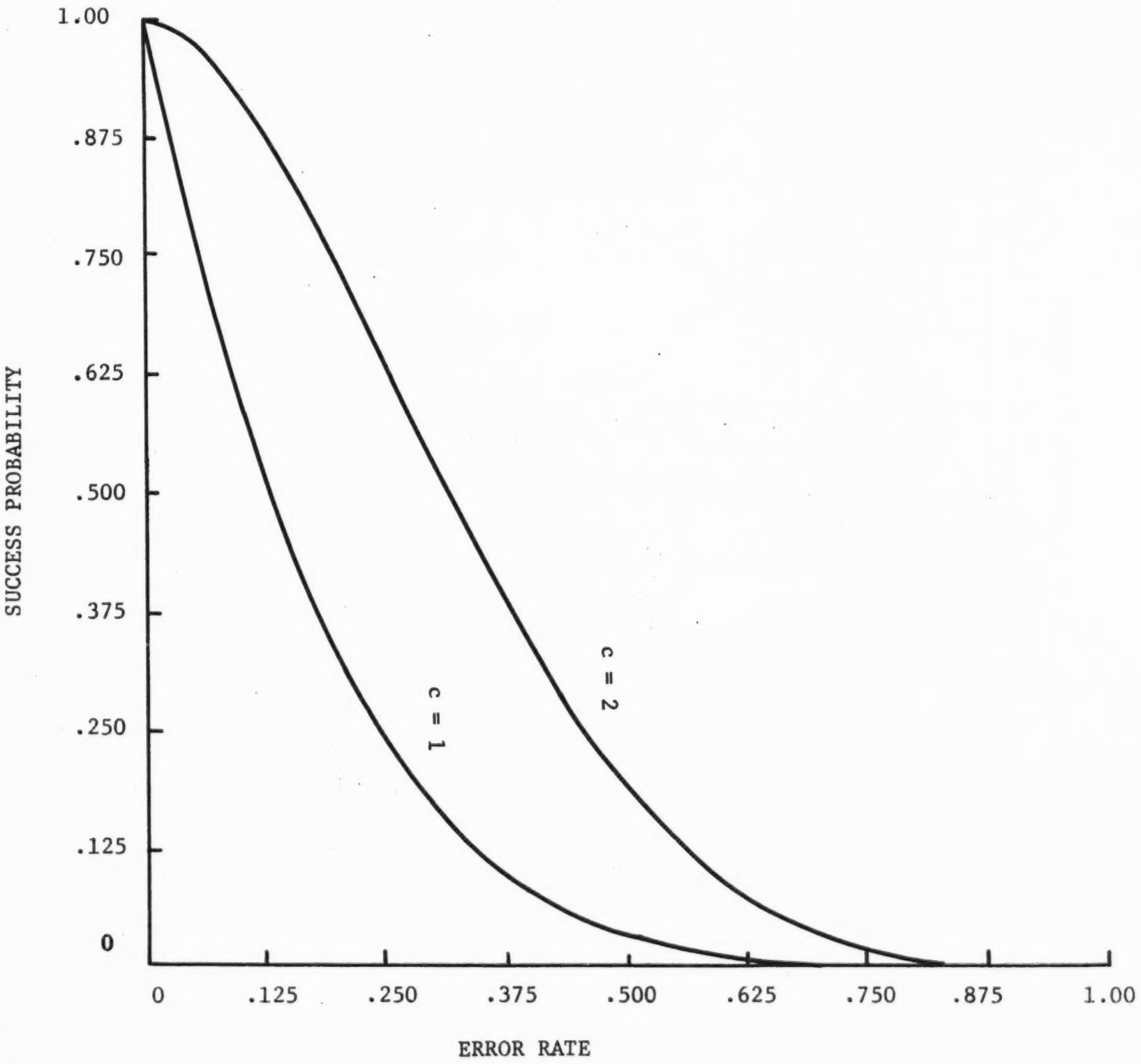
Figure 3



n = 20; c = 4

Lattice Diagram

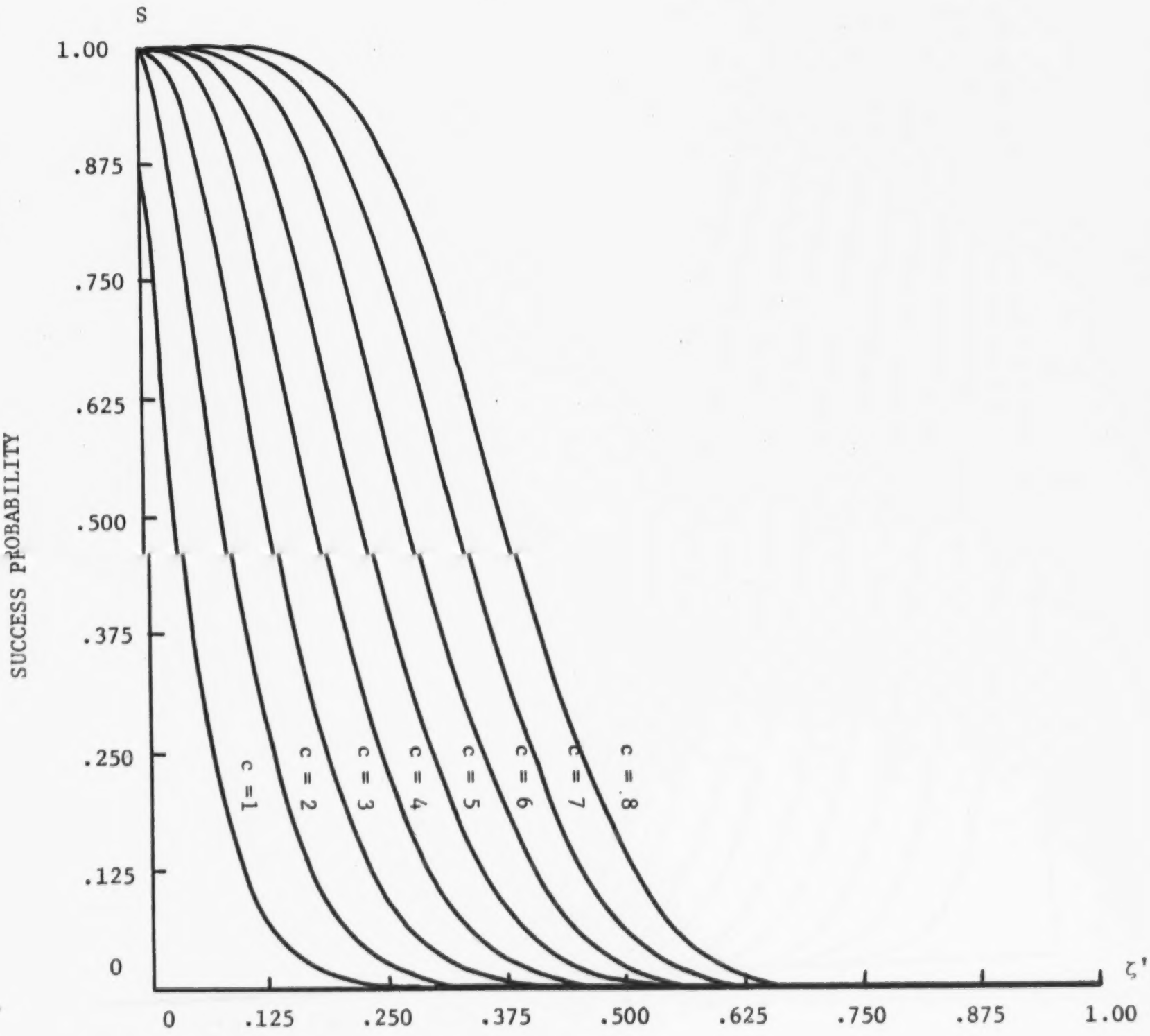
Figure 4



n = 5

OC CURVES FOR FIXED N AND SELECTED CRITERIA

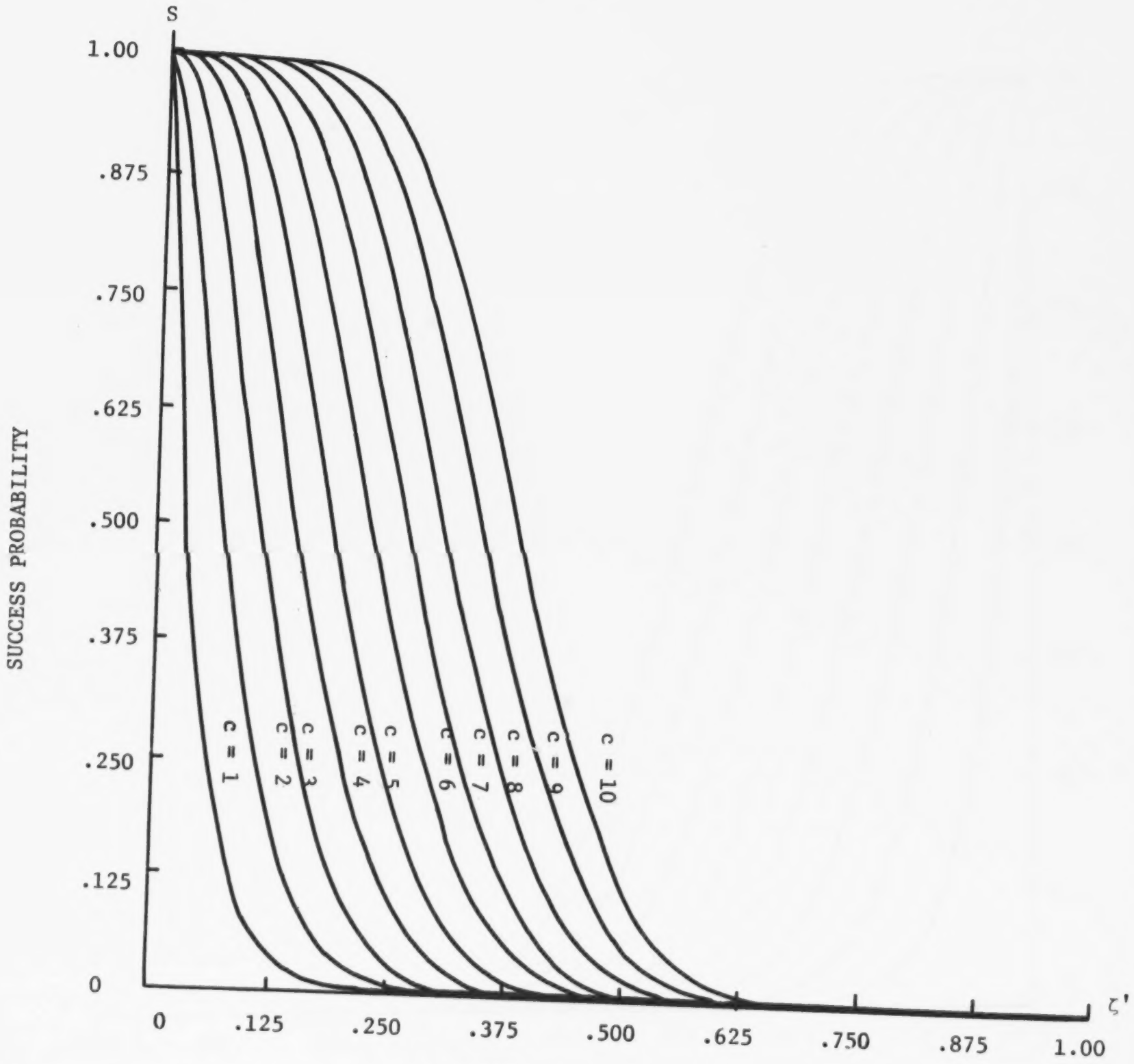
Figure 5



n = 20

OC CURVES FOR FIXED N AND SELECTED CRITERIA

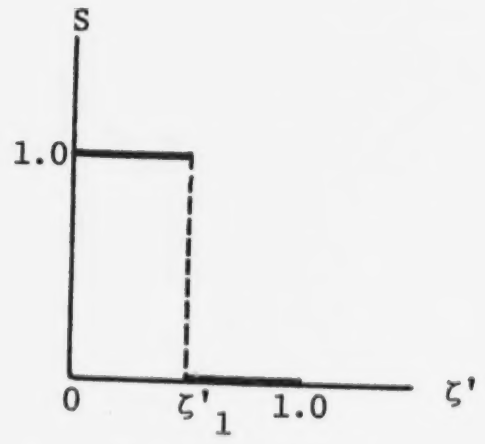
Figure 6



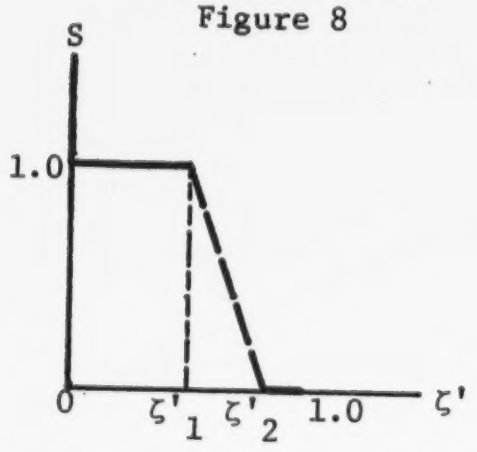
n = 25

OC CURVES FOR FIXED N AND SELECTED CRITERIA

Figure 7



Step-Function OC
Figure 8

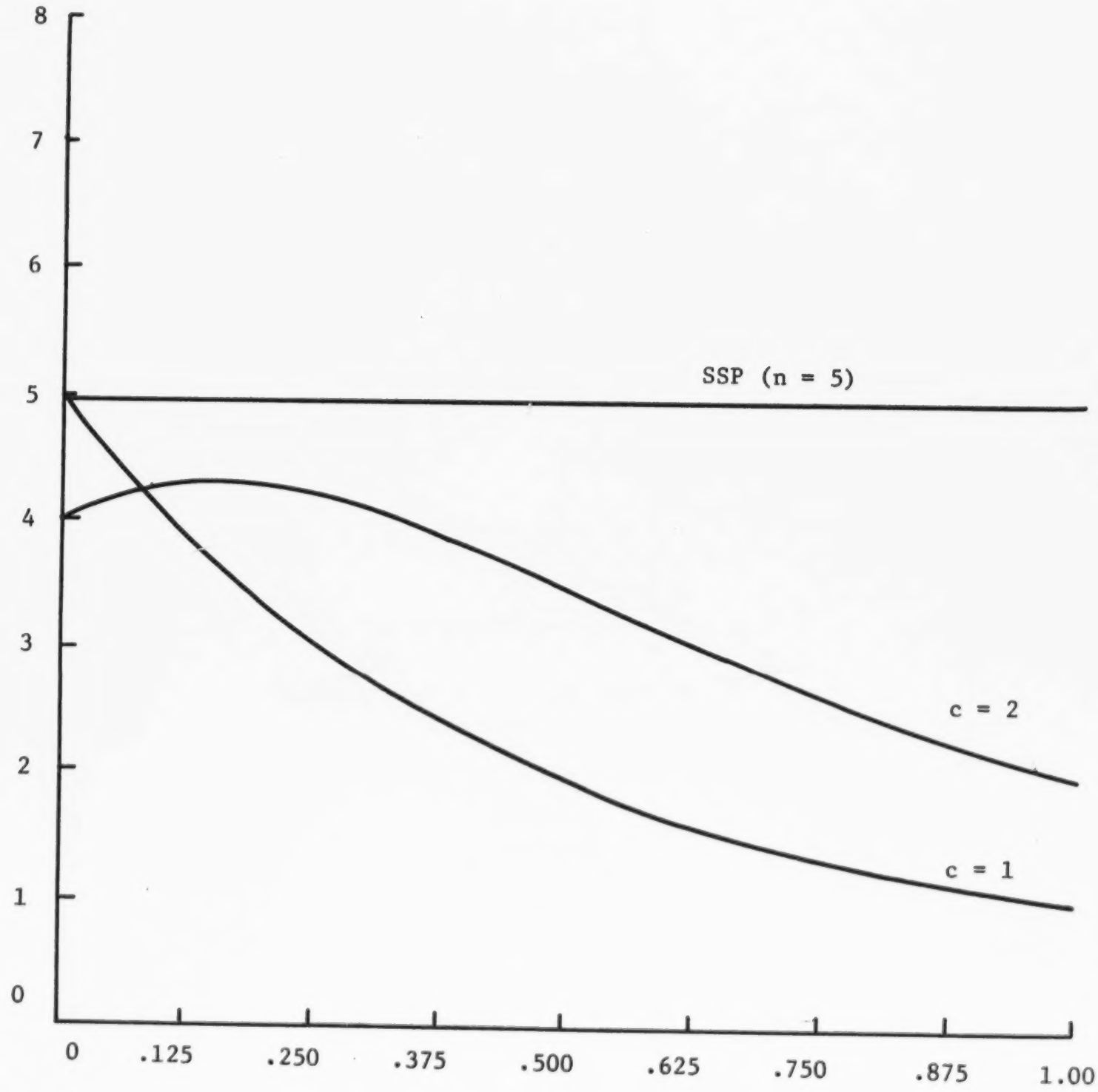


OC with Region of Indifference
Figure 9

True Hypothesis	Accepted Hypothesis H_0	Accepted Hypothesis H_1
H_0	$1 - \zeta$	ζ Type I
H_1	β Type II	$1 - \beta$

Decision Outcome Probabilities

Figure 10

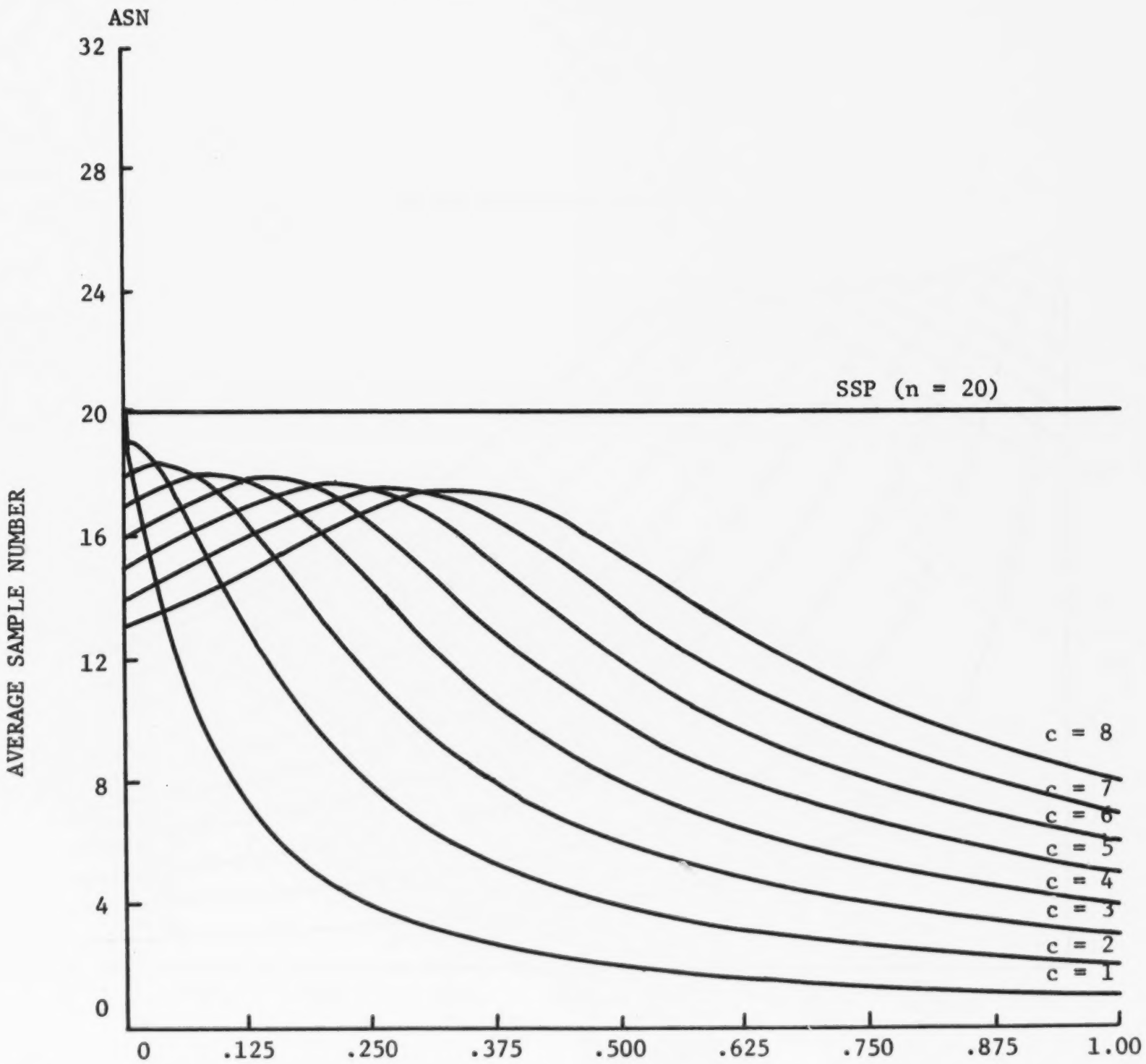


CSSP: n = 5

ASN CURVES FOR FIXED N AND SELECTED CRITERIA

Figure 11

ASN CURVES FOR FIXED N AND SELECTED CRITERIA

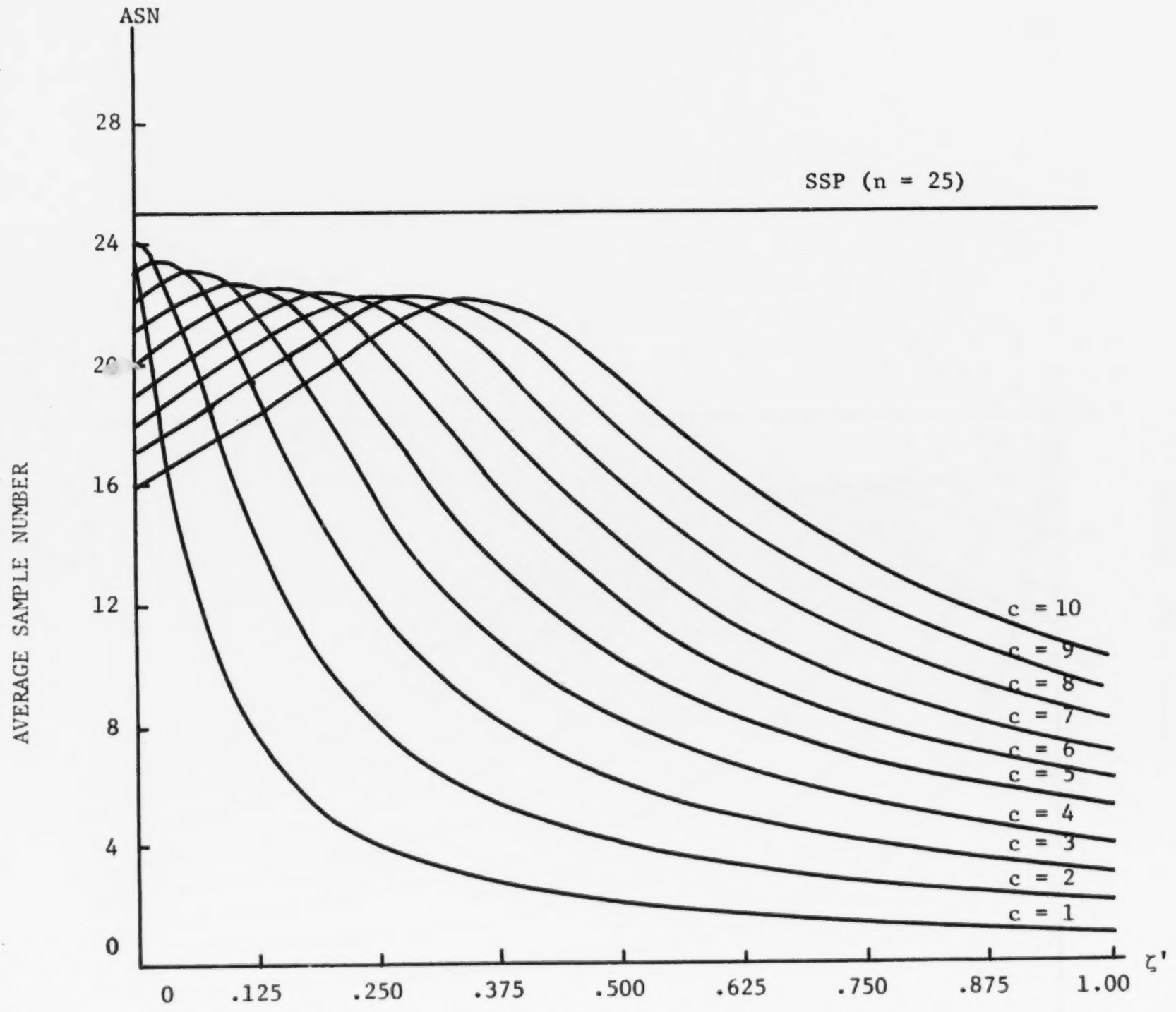


CSSP: n = 20

ASN CURVES FOR FIXED N AND SELECTED CRITERIA

Figure 12

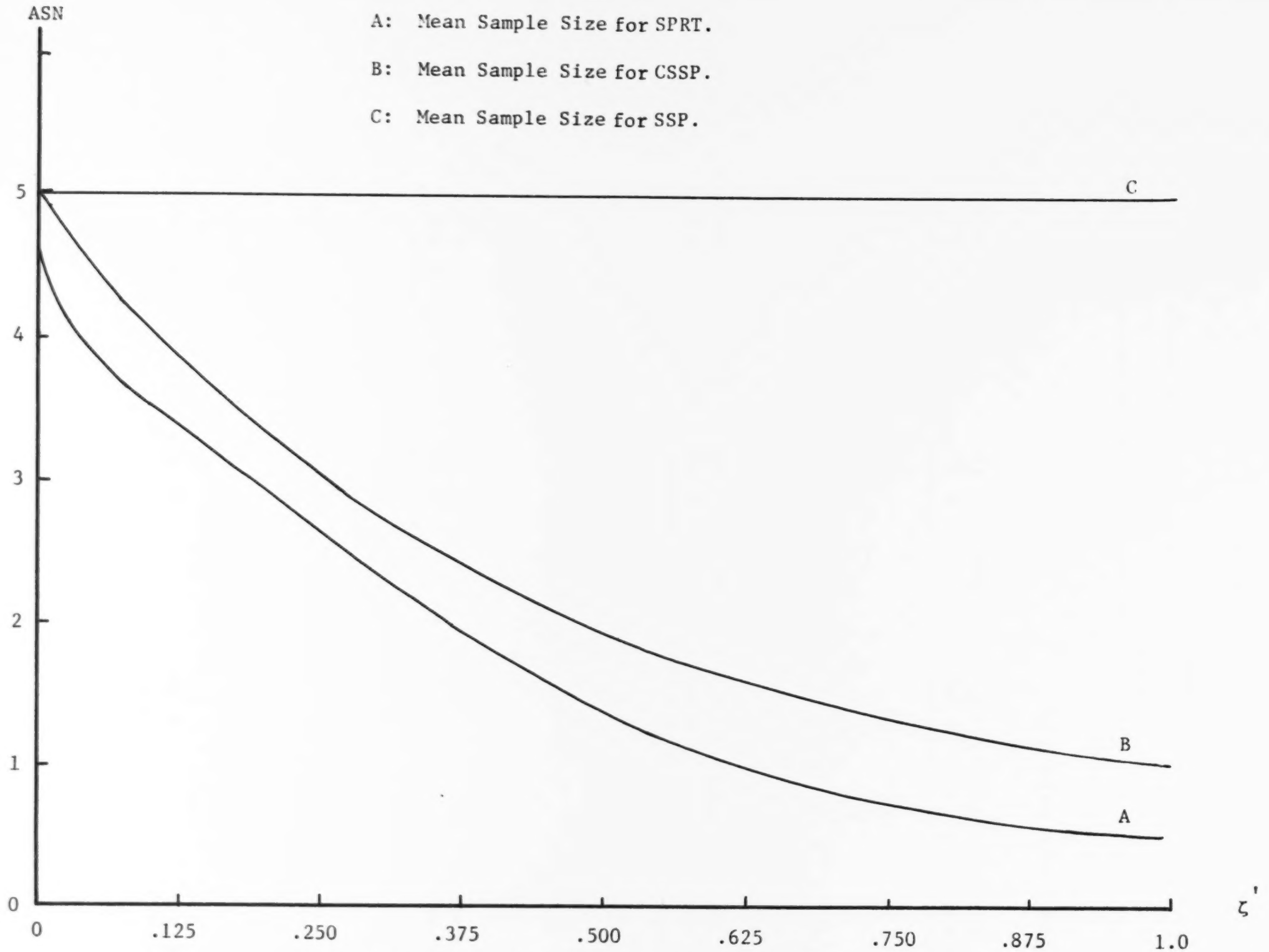
ASN CURVES FOR FIXED N AND SELECTED CRITERIA



CSSP: n = 25

ASN CURVES FOR FIXED N AND SELECTED CRITERIA

Figure 13



COMPARATIVE SAMPLE SIZES FOR VARIOUS SAMPLING PLANS

Figure 14

Random Sample of 2-Digit Multiplication Problems

Generated by Math/CMS

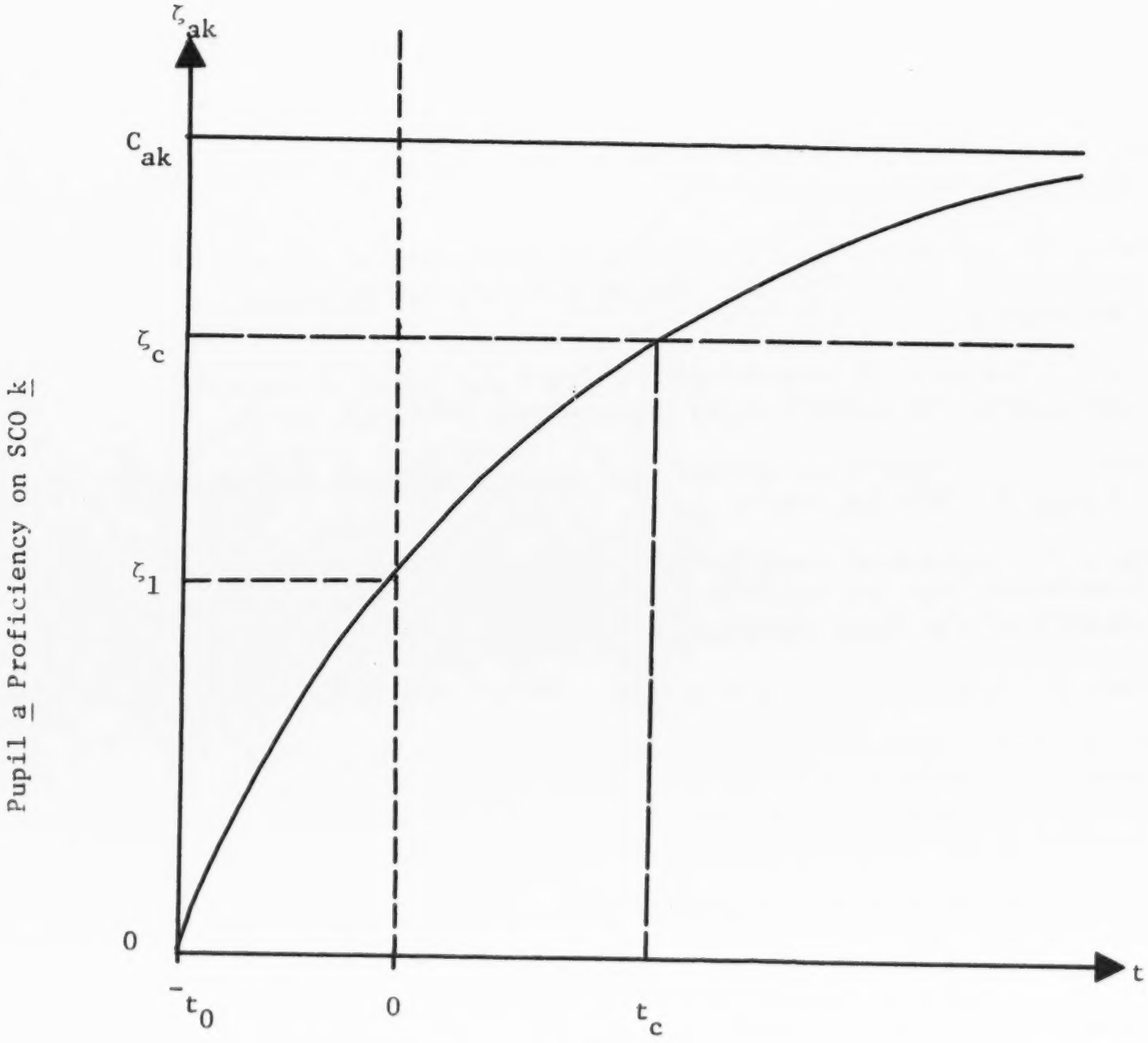
Figure 15

TEST # 082832328 NAME _____

DATE _____

ALL PROBLEMS ARE BASE 10:

1). 21 X 62	2). 35 X 38	3). 15 X 31	4). 65 X 45
5). 71 X 24	6). 62 X 65	7). 26 X 44	8). 31 X 54
9). 22 X 96	10). 25 X 10	11). 61 X 59	12). 23 X 25
13). 33 X 79	14). 83 X 14	15). 89 X 45	16). 11 X 76
17). 66 X 13	18). 83 X 42	19). 39 X 55	20). 86 X 61



Hypothetical Proficiency Development Curve

Figure 16

REFERENCES

Amster, S. J. A modified bayes stopping rule. Annals of Mathematical Statistics, 1963, 34, 1404-13.

Anastasi, A. An empirical study of the applicability of sequential analysis to item selection. Educational and Pyschological Measurement, 1953, 13, 3-13.

Angoff, W. Scales with nonmeaningful origins and units of measure. Educational and Psychological Measurement, 1962, 22, 27-34.

Anscombe, F. J. Sequential estimation. Journal of Royal Statistical Society B, 1953, 15, 1-29.

Armitage, P. Sequential analysis with more than two alternative hypotheses, and its relation to discriminant function analysis. Journal of the Royal Statistical Society B, 1950, 12, 137-44.

Armitage, P. Sequential medical trials. Oxford: Blackwell, 1960.

Armstrong, J. R. Maximizing the information output: A general model for research in education. Paper presented at the Joint Psychometric Conference of University of Wisconsin-University of Chicago, Chicago, October 31, 1966.

Bacon, F. The new organon and related writings. New York: Liberal Arts Press, 1960.

Bartell, E. A study of feasibility of an individualized instructional program in elementary school mathematics. Unpublished doctoral dissertation, University of Wisconsin, 1965.

Becker, J. W. Toward automated learning. Paper presented at the AERA annual meeting, Chicago, February 8, 1968.

Beckhofer, R. E. A sequential multiple-decision procedure for selecting the best one of several normal populations with a common unknown variance, and its use with various experimental designs. Biometrics, 1958, 14, 408-29.

Beckhofer, R. E. A single sample multiple-decision procedure for ranking means of normal populations with known variances. Annals of Mathematical Statistics, 1954, 25, 16-39.

Bloom, B., et al The classification of educational goals: Cognitive domain. New York: David McKay, 1956.

Bloom, B. S. Learning for mastery. Evaluation comment, 1968, 1 (2).

Bormuth, J. R. On the theory of achievement test items. (unpublished manuscript, University of Chicago, 1968.)

Brownell, W. and Moser, E. H. Meaningful vs. mechanical learning: A study in grade III subtraction. Duke University Research Studies in Education, 1948, 321-37.

Bruner, J., Goodnow, J. and Austin, G. A study of thinking. New York: Science Editions, 1965.

Brueckner, L. and Grossnickle, F. How to make arithmetic meaningful. New York: Winston, 1947.

Bush, R. R. and Mosteller, F. Stochastic models for learning. New York: Wiley, 1955.

Cambridge Conference on School Mathematics. Goals for school mathematics. Boston: Houghton Mifflin, 1963.

Campbell, D. T. and Stanley, J. C. Experimental and quasi-experimental designs for research on teaching. The Handbook of Research on Teaching, 1963, 171-246.

Carnap, R. Logical foundations of probability. Chicago: University of Chicago Press, 1950.

Carne, E. B. Artificial intelligence techniques. London: Macmillan, 1965.

Carroll, J. B. A model of school learning. Teachers College Record, 1963, 64 (8).

Carroll, J. B. The predictors of success in intensive language training. In R. Glaser (Ed.), Training Research and Education. Pittsburgh: University Pittsburgh Press, 1962, 87-136.

Cocker, W. Cocker's Arithmetic. London, 1678.

Cogswell, J. F., Egbert, R. L., Marsh, D. G., and Yett, F. A. Construction and use of the school simulation vehicle. Technical Memorandum TM-2084, System Development Corporation, Corporation, Santa Monica, California, 1964, 17.

College Entrance Examination Board. The report of the commission on mathematics, Vol. I. Princeton, N. J.: Educational Testing Service, 1959.

Commission on Mathematics of the Progressive Education Society. Mathematics in general education, (PEA Report). New York: Appleton-Century, 1940.

Congreve, W. J. Independent learning. North Central Association Quarterly, 1965, 40, 222-228.

Cornfield, J. and Tukey, J. Average values of mean squares in factorials. Annals of Mathematical Statistics, 1956, 27.

Coulson, J. and IMS Project Staff. Progress report for the instructional management system. Technical Memorandum TM-3298/00, System Development Corporation, Santa Monica, California, May 1968.

Coulson, J. and Cogswell, J. Effects of individualized instruction on testing. Journal of Educational Measurement, 1965, 2.

Cox, R. and Vargas, J. A comparison of item-selection techniques for norm-referenced and criterion-referenced tests. Learning Research and Development Center, University of Pittsburgh, 1966. (ERIC Document 010 517).

Cremins, L. The transformation of the school. New York: Alfred A. Knopf, 1961.

Cruickshank, D. R. Simulation: new direction in teacher preparation. Phi Delta Kappan, 1966, 48, 23-4.

Darlington, R. and Stauffer, G. Use and evaluation of discrete test information in decision making. Journal of Educational Psychology, 1966, 50, 125-129.

Davidson, D. and Suppes, P. Decision making: An experimental approach. Stanford, California: Stanford University Press, 1957.

Davis, R. B. A modern mathematics program as it pertains to the interrelationship of mathematical content, teaching methods and classroom atmosphere, (The Madison Project). Cooperative Research Project D-093, 1965.

Dettre, J. R. Video-taping simulated teaching: A tool in general methods. Audiovisual Instruction, 1967, 12: 693-5.

DeVault, M. Vere. What is mathematics curriculum research? The Arithmetic Teacher, 1966, 636-639.

DeVault, M. Vere and Kriewall, T. Perspectives in elementary school mathematics. Columbus, Ohio: Merrill, 1969.

Ebel, R. L. Content standard scores. Educational and Psychological Measurement, 1962, 22, 15-25.

Ebel, R. L. The use of response time measurements in the construction of educational achievement tests. Educational and Psychological Measurement, 1953, 13, 391-401.

Ebel, R. L. The value of internal consistency in classroom examinations. Journal of Educational Measurement, 1968, 5, 71-73.

Eberhart, J. It's just a game: GREMEX, Goddard research engineering management exercise. Science News, 1966, 90, 256.

Estes, W. K. and Suppes, P. Foundations of statistical learning theory, Volume II. Palo Alto, California: Stanford University Press, 1959.

Eby, F. The development of modern education. Englewood Cliffs, N.J.: Prentice-Hall, 1952.

Feigenbaum, E. A. and Feldman, J. (Eds.) Computers and thought. New York: McGraw-Hill, 1963.

Flanagan, J. Developing a functioning model of an educational system for the '70's. Palo Alto, California: American Institute for Research, 1967.

Flanagan, J. Program for learning in accordance with needs. Paper presented at American Educational Research Association, Chicago, 1968.

Flanagan, J. Discussion. Educational and Psychological Measurement, 1962, 22, No. 1, 35-37.

Flanagan, J. Scaled scores. The Cooperative Test Service of the American Council on Education, 1939.

Flanagan, J. C. Units, scores, and norms. In E. F. Lindquist (Ed.), Educational Measurement. Washington, D. C.: American Council on Education, 1950.

Flexman, R. and Horowitz, M. W. Games businessmen play. Time, 1966, 88 (111).

Fogel, L. J., Owens, A. J. and Walsh, M.J. Artificial intelligence through simulated evolution. New York: Wiley, 1966.

Forgy, E. Cluster analysis of multivariate data vs. interpretability of classifications. Paper presented to WNAR meetings, University of California, Riverside, 1965.

Gardner, E. F. Normative standard scores. Educational and Psychological Measurement, 1962, 22, 7-14.

Gerlach, V. S. Academic games in simulation of instruction. Audiovisual Instruction, 1967, 12, 609-10.

Glaser, R. Instructional technology and the measurement of learning outcomes: Some questions. American Psychologist, 1963, 18, 519-521.

Groen, G. J. and Atkinson, R. C. Models for optimizing the learning process. Psychological Bulletin, 1966, 66, 309-20.

Gulliksen, H. Theory of mental tests. New York: Wiley, 1950.

Hammock, J. Criterion measures: Instruction vs. selection research. Paper read at the meeting of the American Psychological Association, September 1960.

Hansen, R. and Schwarz, G. An investigation of computer-based science testing. Report of a project submitted to the College Entrance Examination Board, 1968.

Halperin, M., Hartley, H., and Hoel, P. Recommended standards for statistical symbols and notation. American Statistician, 1965, 19, 12-14.

Hill, I. D. The economic incentive provided by sampling inspection. Applied Statistics, 1960, 69-81.

Hively, W. II. Generalizability of performance by job corp trainees on a universe-defined system of achievement tests in elementary mathematics calculation. Paper presented at the annual meeting of the American Educational Research Association, Chicago, February 1968.

Hunt, E. G., Marin, J., and Stone, A. Experiments in induction. New York: Academic Press, 1966.

Karush, W. and Dear, R. E. Optimal stimulus presentation strategy for a stimulus sampling model of learning. Scientific Paper SP-1869. Santa Monica, California: Systems Development Corporation, 1964.

Koerner, J. D. The miseducation of American teachers. Baltimore: Penquin Books, 1963.

Kopstein, F. F. and Seidel, R. J. Computer-administered instruction versus traditionally administered instruction: Economics. Audiovisual Communication Review, 1968, 16, 147-175.

Lord, F. and Novick, M. Statistical theories of mental test scores. Reading, Mass.: Addison-Wesley, 1968.

MacDonald, J. B. Curriculum theory: Problems and a prospectus. Paper presented at the Professors of Curriculum Meeting, Miami Beach, Florida, 1964.

MacDonald, J. B. Myths about instruction. Educational Leadership, May 1965.

MacDonald, J. B. Structures in curriculum. Speech delivered at the University of Wisconsin-Milwaukee, May 23, 1966.

Magnusson, D. Test theory. Reading, Mass.: Addison-Wesley, 1966.

Markle, D. The development of the Bell system first aid and personal safety course: An exercise in the application of empirical methods to instructional system design. Final Report (AIR-E81-4167-FR), American Institutes for Research, Palo Alto, 1967.

Mathematical Association of America. The reorganization of secondary school mathematics. Washington, D. C.: The Association, 1923.

Monroe, W. S. Development of arithmetic as a school subject. Bulletin No. 10, Washington, D. C.: U. S. Government Printing Office, 1917.

Morganstern, O. and Von Neumann, J. Theory of games and economic behavior. Princeton: Princeton University Press, 1944.

National Council of Teachers of Mathematics. The place of mathematics in secondary education (15th yearbook). Washington, D. C. : The Council, 1940.

National Council of Teachers of Mathematics. The teaching of arithmetic (10th yearbook). Washington, D. C.: The Council, 1935.

Oettinger, A. The myths of educational technology. Saturday Review, May 18, 1968.

Platt, J. Strong inference. Paper presented to the American Chemical Society, September 1963.

Popham, W. J., and Husek, T. R. Implications of criterion-referenced measurement. Unpublished manuscript. Southwest Regional Laboratory, 1968.

Press, L. I., and Rogers, M. S. IDEA - a conversational, heuristic program for inductive data exploration and analysis. Paper presented at the annual meeting of the Association of Computing Machinery, 1967.

Radio Corporation of America. Instructional 70 : User's guide to instructional language-1. Palo Alto, California: RCA Instructional Systems, 1967.

Rice, J. The public school system of the United States. New York: Century, 1893.

Romberg, T. A note on multiplying fractions. The Arithmetic Teacher, 1968, 15.

Romberg, T., and DeVault, M. V. Mathematics curriculum: Needed research. Journal of Research and Development in Education, 1967, 1, 95-110.

Runnels, L. K., et al. Near perfect runs as a learning criterion. Journal of Mathematical Psychology, 1968, 5, 362-368.

Siegel, S. Nonparametric statistics. New York: McGraw-Hill, 1956.

Smallwood, R. Quantitative methods in computer-directed teaching systems. Final report on ONR Contract Nonr-255(84), Stanford University, March 1967.

Spector, B. Of man, machines and education. Automated Education Letter, 1965, 1 (2), 3-12.

Stein, K. J. New simulators offer wider data range. Aviation Week, 1967, 87 (79),

Statistics Research Group. Sequential analysis of statistical data applications. New York: Columbia University Press, 1945.

Suppes, P., Jerman, M., and Brian, D. Computer-assisted instruction: Stanfords' 1965-66 arithmetic program. New York: Academic Press, 1968.

Thorndike, E. L. The psychology of arithmetic. New York: MacMillan, 1922.

Thorndike, E. L., and Woodworth, R. S. The influence of improvement in one mental function upon the efficiency of other functions. Psychological Review, VIII, 1901.

Von Braun, W. Practicing a moon landing on earth; training with lunar landing research vehicle. Popular Science, 1966, 188, 98-9.

Walbesser, H., and Carter, H. Differences in group and individually administered tests of the same behavior. Paper presented at AERA meeting, Los Angeles, February 1969.

Wald, A. Sequential analysis. New York: Wiley, 1947.

Wald, A. Statistical decision functions. New York: Wiley, 1950.

Wald, A., and Wolfowitz, S. Optimum character of the sequential ratio probability test. Ann Math Stat, 1948, 19, 326-39.

Walker, H. M., and Lev, J. Elementary statistical methods. New York: Holt, Rinehart, and Winston, 1958.

Washburne, C. W. Mental age and the arithmetic curriculum. Journal of Education Psychology, 1931, 23, 210-31.

Wetherill, G. B. Sequential methods in statistics. New York: Wiley, 1966.

Wilson, G. M. The social utility theory as applied to arithmetic. Its research basis, and some of its applications. Journal of Educational Research, 1948, 321-37.

Wright, B. D. Sample-free test calibration and person measurement. Paper presented at The Invitational Conference on Testing Problems, October, 1967.